AtomosVision



http://www.atomos.it

Indice

Parte I

1 Help on line AtomosVision

AtomosVision è lo strumento per la realizzazione di report di analisi dei dati.

AtomosVision permette la creazione di report (navigabili o statici per finalità di stampa) a partire da dati estratti da database attraverso query customizzabili.

Parte II

2 Menu File

2.1 Database

2.1.1 Connessione

All’apertura, il sistema si connette al Data Base di default; tramite questa voce è possibile scegliere un'altra connessione tra quelle precedentemente create.

La connessione di default deve chiamarsi DEFAULT_ATV.CON.

2.1.2 Nuova connessione

Permette all’utente di creare nuove connessioni a Data Base esistenti, specificando Alias, User Name e Password.

2.1.3 Carica tabelle

Una volta lanciato, il sistema si connette automaticamente al Data Base di default e carica le tabelle in memoria come definite nel file ATOMOSVISION.CDB.

L'applicazione funziona correttamente anche se il file CDB non è presente o è vuoto.

Per aggiornare i dati caricati in memoria, in seguito a modifiche avvenute sul database fisico, lanciare manualmente il caricamento delle tabelle tramite l’apposita voce di menu.

2.2 Utente

2.2.1 Login as

Con questo comando è possibile utilizzare il programma con un utente ACN diverso da quello attuale.

2.2.2 Impersonifica utente/nodo

Questa utilità permette di salvare le modifiche fatte (report, configurazioni,...) ad un livello differente rispetto a quello dell'utente loggato. Per poter accedere a questa funzione l'utente deve avere in ACN un livello >= 90 (il livello utente è un numero compreso tra 0 e 99 associato all'utente in ACN).

L'impersonificazione DEVE essere fatta PRIMA di modificare i report e comporta la chiusura di tutti i report aperti.

L'utente che ha effettuato l'impersonificazione mantiene le sue licenze. (non acquisisce quelle dell'utente su cui si è "trasferito").

L'impersonificazione può essere fatta verso un NODO o verso un altro UTENTE.

L'impersonificazione verso un UTENTE consente di accedere ai livelli della gerarchia esattamente come se si trattasse dell'utente stesso. Chi impersonifica vede anche la cartella personale (quella del vecchio standard Nicim) dell'utente che viene impersonificato.

Durante l'impersonificazione verso un NODO invece l'utente mantiene l'accesso alla propria cartella personale (quella del vecchio standard Nicim).

La struttura gerarchica degli utenti tramite la creazione di nodi deve essere precedentemente effettuata tramite il comando "Company chart" di ACN.

2.3 Modalità configuratore

Modalità selezionabile solo da parte di utenti con la licenza di operatività completa.
Se si intende iniziare una sessione di modifica dei layout, occorre selezionare questa modalità. Diversamente, le eventuali "piccole

modifiche" fatte ai report sono quelle che può fare qualsiasi utente senza licenze particolari e con un livello in ACN >= 50 tramite il comando che è sulla toolbar del singolo strumento grafico e sono memorizzabili tramite il comando (solo a livello del singolo utente corrente).

La modalità configuratore è accessibile solo se l'utente in ACN ha un livello >= 90. Il livello utente è un numero compreso tra 0 e 99 associato all'utente in ACN.

Gli utenti con un livello < 50 non possono accedere alla modalità configuratore ne' possono salvare alcuna modifica ai report, neanche a livello locale.

Accedendo alla modalità configuratore tutti i report aperti verranno chiusi.

2.4 Esci

Termina l'applicazione.

Parte III

3 Menu Analisi

Questo menu contiene l'elenco dei report di analisi (layout di indagine) definiti nel sistema.

I report contrassegnati dall'icona sono report salvati localmente ed accessibili solo dall'utente corrente, i report contrassegnati dall'icona sono report ereditati dal company chart.

3.1 Organizza questo menu

Questa opzione permette di costruire e organizzare il menù in modo personalizzato.
La struttura del menu è visualizzata ad albero, con un numero di livelli personalizzabile. Posizionandosi su un nodo dell'albero, infatti è possibile, con un click destro del mouse, scegliere l'opzione di "Crea cartella" che genera un nuovo nodo allo stesso livello della voce selezionata, a cui è possibili dare il nome che si preferisce.
Gli item possono poi essere sposati tra le cartelle con drag&drop.
Nell'elenco compaiono tutti i report presenti nel sistema (ereditati dal company chart).

La configurazione del menu è accessibile solo quando l'applicazione è in Modalità Configuratore e viene sempre memorizzata nel livello del company chart a cui appartiene l'utente.

Cliccando sul pulsante di "Reset" la configurazione locale viene resettata a favore della configurazione del menu ereditata dalla gerarchia del company chart.

Parte IV

4 Menu Stampe

Questo menu contiene l'elenco dei report per la stampa definiti nel sistema.

I report di stampa sono costruiti sfruttando la tecnologia di FastReport.

4.1 Organizza questo menu

Vedi Organizza questo menu

Parte V

5 Menu Funzioni

Questo menu contiene l'elenco di script che possono eseguiti all'interno dell'applicazione.

Gli script devono essere scritti in linguaggio pascal script. Una breve guida a questo linguaggio è disponibile qui: http://necago.ic.cz/prj/pascalscript/doc/PascalScript-lang.pdf.

5.1 Organizza questo menu

Vedi Organizza questo menu

Parte VI

6 Menu Strumenti

6.1 Configurazione stili

Tramite questo pannello è possibile definire gli attributi grafici dell'applicazione: proprietà dei font dei vari oggetti, colori di righe, di griglie e tabelle, ecc…

6.2 Descrizione Colonne / Dizionario

Descrizione Colonne

Questo strumento permette di definire dei default per le i campi nei report dell'applicazione.

Ovvero, quando si crea un report appartenente ad un certo Contesto, esso verrà generato utilizzando questi default, che poi, potranno, report per report, essere ovviamente modificati.

I vari campi (si pensi ad esempio a "CODICE_PARTE") si possono trovare su più report, ovvero in più contesti. Pertanto si avranno in questo elenco, più righe per uno stesso campo: una senza codice contesto, che indica il default per tutti i contesti, e più righe, una per ogni contesto che la utilizza.

Il campo può essere personalizzato diversamente per i vari contesti, oppure, se si vuole effettuare una modifica valida per tutti i contesti, occorre modificare la riga con contesto vuoto.

Di un'etichetta si può modificare:

- Etichetta: ovvero l'intestazione della colonna/campo in cui è rappresentato il campo
- Formato: ovvero la rappresentazione del valore. (Si pensi ad esempio alla rappresentazione dei decimali in un numero)
- Visibilità: ovvero se la colonna/campo subito viene per default creata o meno visibile

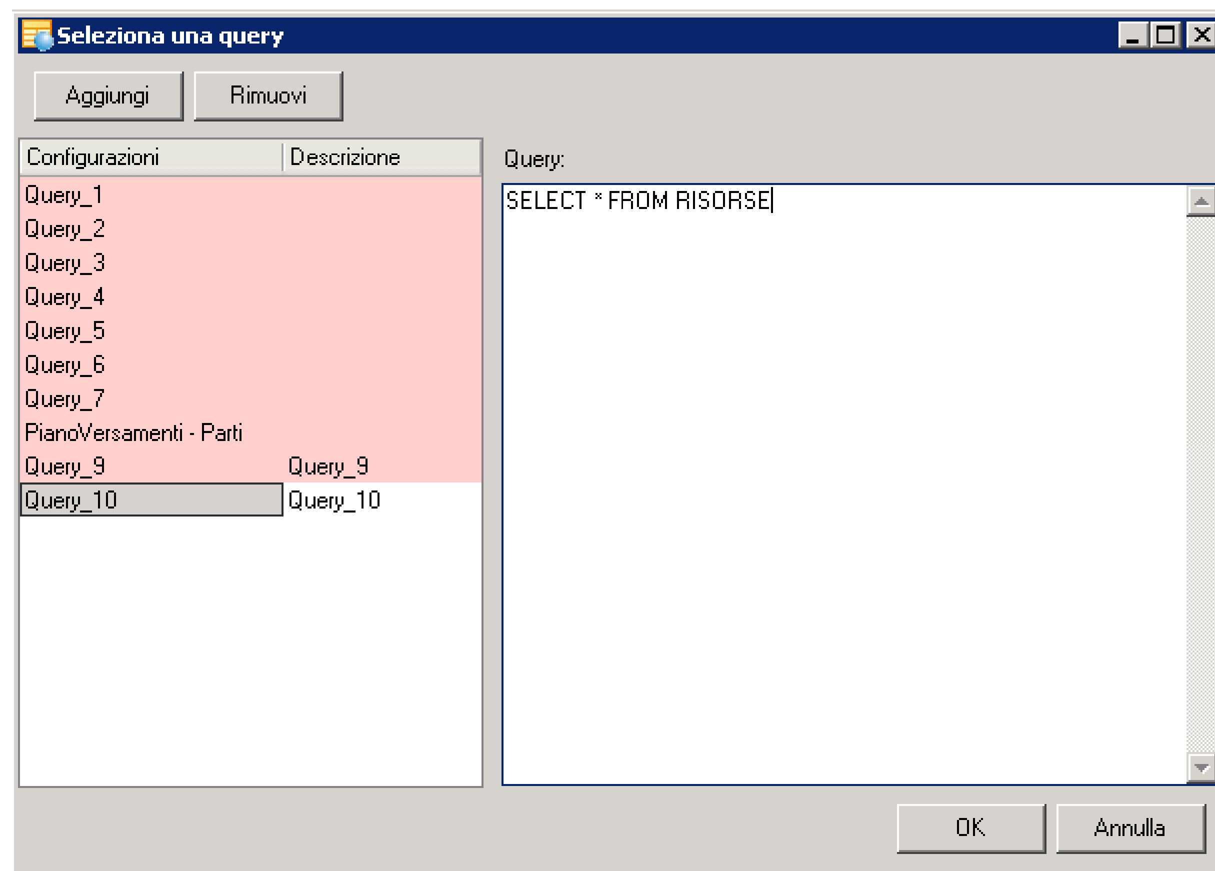
Dizionario

A livello di utente ACN è possibile indicare una lingua tra quelle previste, e di conseguenza l'applicazione visualizzerà tutte le stringhe nella lingua prescelta.
In questo caso, però, se sono state modificate delle Etichette nella versione italiano, occorre modificare tale Etichetta anche nelle altre lingue utilizzate, inserendo una traduzione premendo l'icona della matitina, che si trova accanto ad ogni Etichetta: questa viene riportata nel dizionario e lì è possibile scriverne la traduzione in una o più delle lingue previste.

6.3 Query

6.3.1 Query Manager

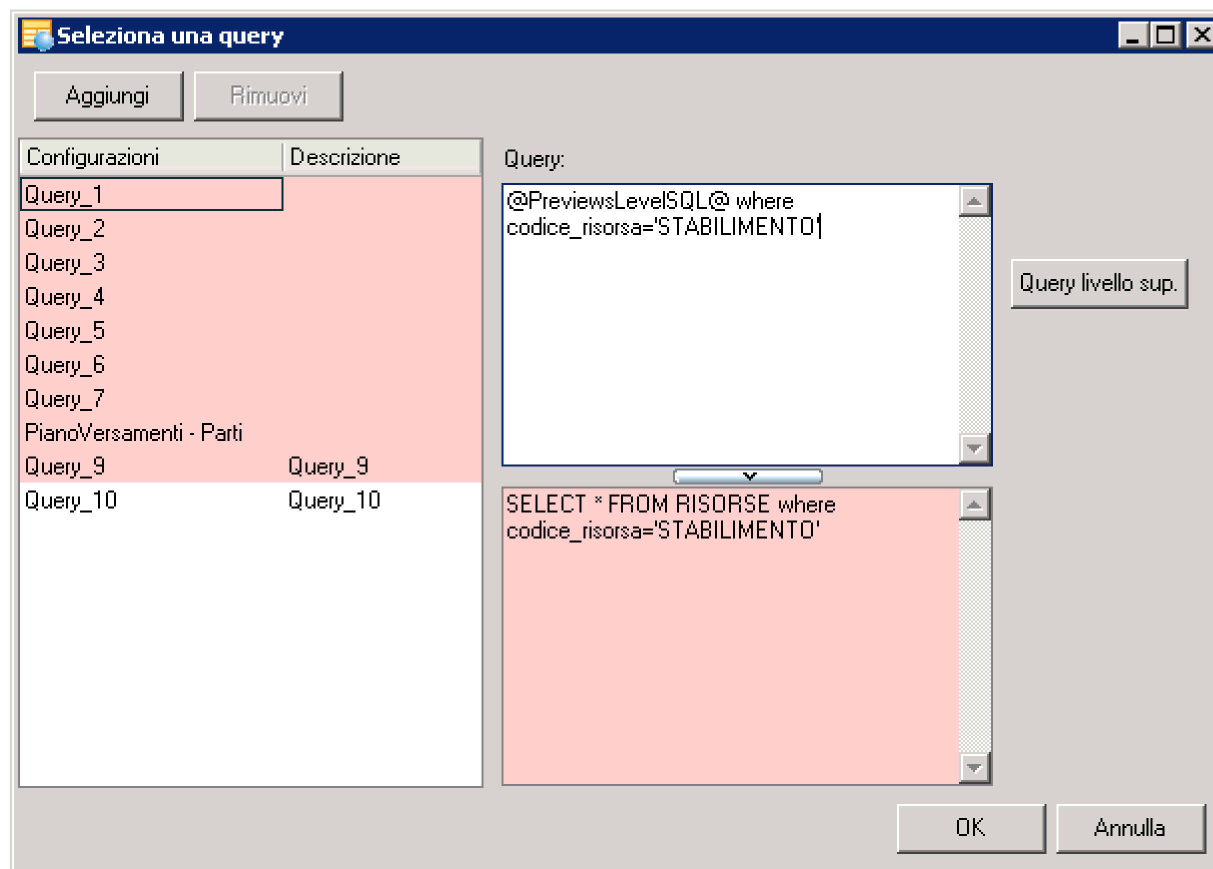
Il query manager presenta l'elenco delle query definite all'interno del sistema di reportistica. Le query sono utilizzate per alimentare griglie, pivot e quant'altro appartenenti ai report di analisi.
In questo pannello l'utente, oltre a prendere visione di tutte le query presenti, può aggiungerne di nuove o effettuare delle modifiche a quelle esistenti.
Quando una query viene creata (pulsante "Aggiungi"), viene assegnata al livello del company chart a cui appartiene l'utente corrente e ivi salvata per il suo utilizzo. L'utente può modificare o cancellare (pulsante "Rimuovi") le query appartenenti al suo livello di company chart liberamente.



Se una query appartiene ad un livello superiore a quello dell'utente corrente, nella maschera apparirà con lo sfondo rosa. In questo caso la query non potrà essere ne' eliminata ne' rinominata dall'utente.

L'utente potrà comunque crearne un **override**, ovvero una riscrittura, a beneficio dei report del proprio livello.

L'override potrà interessare una riscrittura integrale della query o una modifica parziale, magari per aggiungere una nuova where condition.

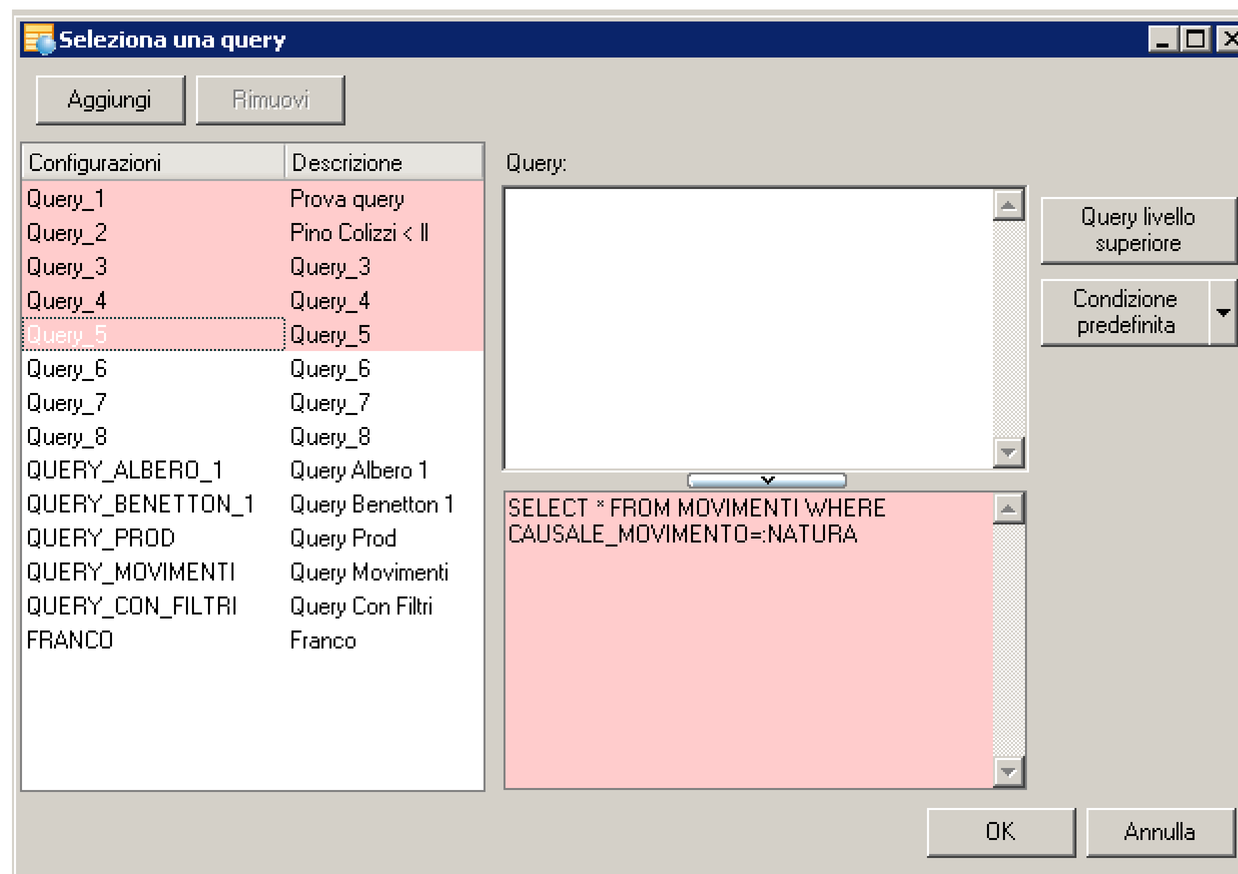


Nell'esempio è stata selezionata dall'utente la Query_1. Cliccando sul pulsante "Query livello sup." nell'area di modifica viene aggiunto il tag @PreviewsLevelSQL@ che rappresenta il testo della query originale. L'utente aggiunge la stringa "where codice_risorsa='STABILIMENTO'", nel box rosa in basso il risultato della riscrittura.

Questa riscrittura permette all'utente di aggiungere condizioni specifiche alle query valide per un certo nodo del company chart senza dover riscrivere i report di analisi definiti ad un livello più alto.

Durante la scrittura di una query l'utente può inserire uno o più **parametri**. Un parametro è un valore che verrà richiesto all'utente ogni volta che la query verrà eseguita (tipicamente all'apertura di un report di analisi) per condizionarne l'esecuzione. Il parametro è costituito da una stringa preceduta dal carattere dei 2 punti (:).

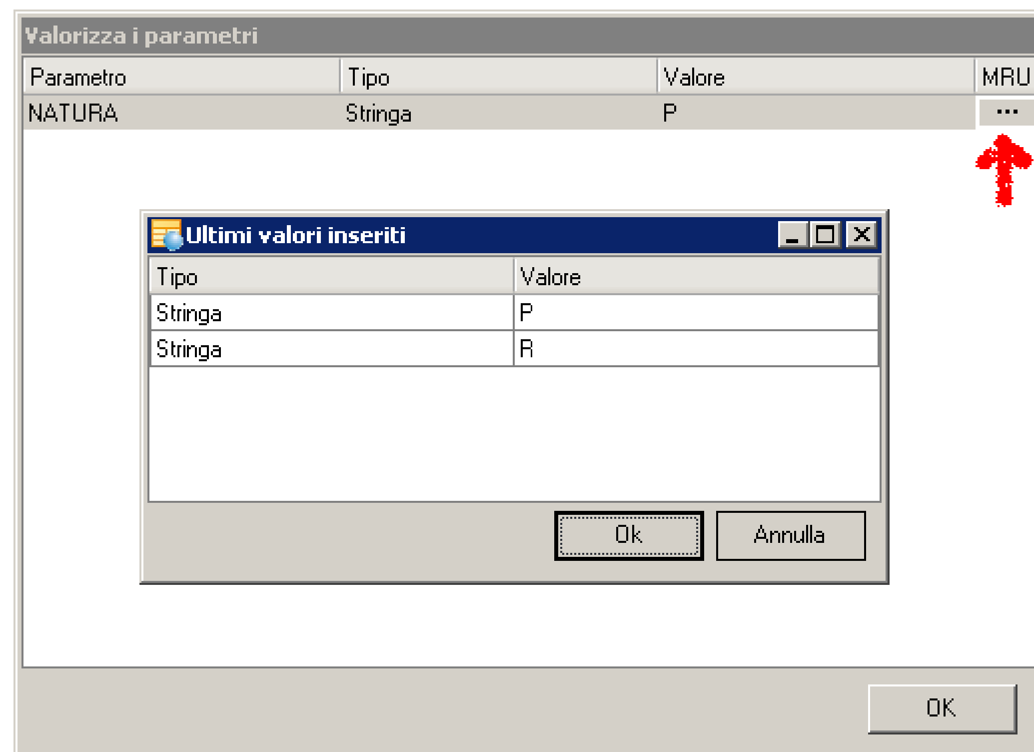
Attenzione: in generale bisogna evitare di inserire commenti nella query che al loro interno contengano stringhe precedute dai 2 punti (:). Il parsing della query per estrarre i parametri da valorizzare potrebbe infatti essere ingannato dal contenuto del commento ed incorrere in malfunzionamenti.



Nell'esempio l'utente ha parametrizzato la query Query_5 aggiungendo un parametro :NATURA.

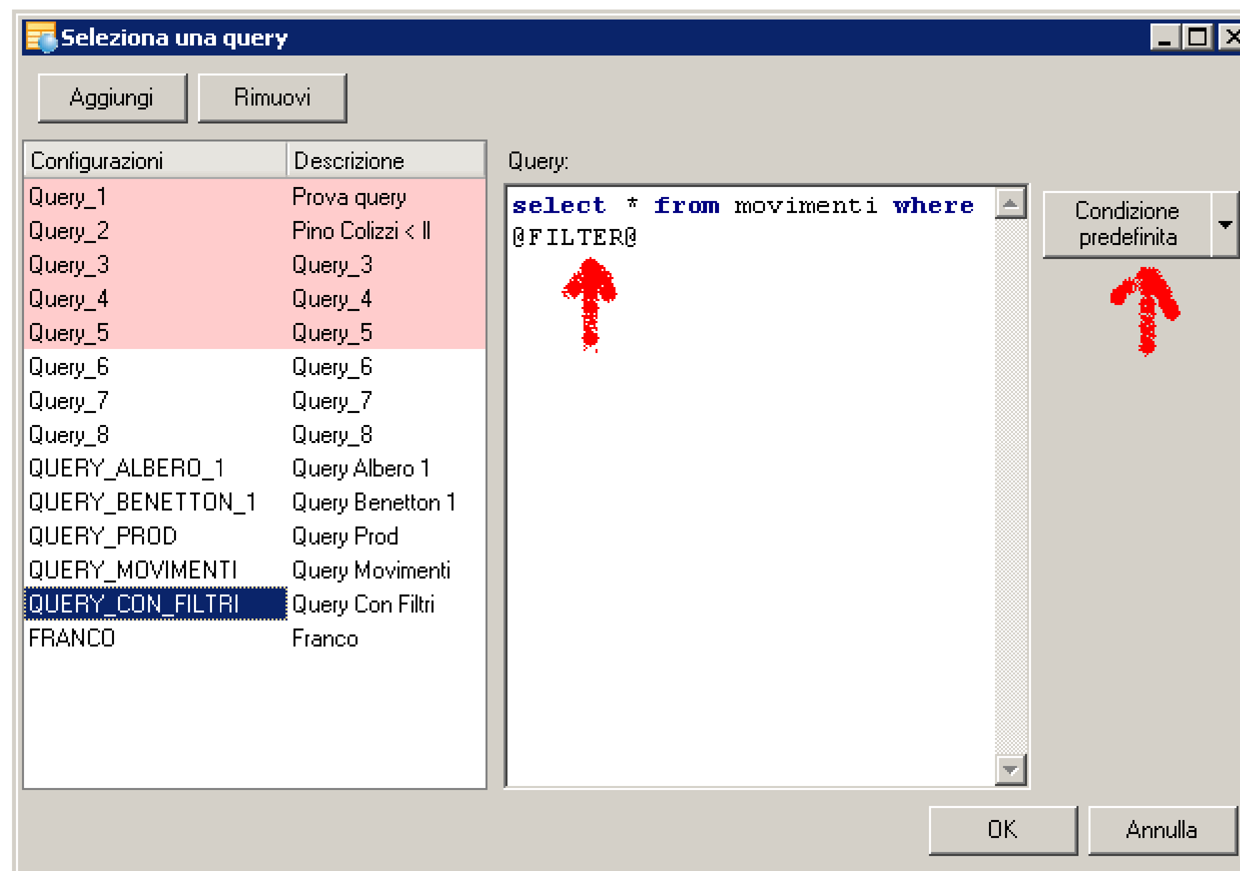
All'apertura di un report che utilizza la Query_5 per alimentare, ad esempio, una griglia, una maschera di inserimento viene proposta. In questa maschera l'utente può definire il tipo di parametro (stringa, intero, ecc..) e assegnargli un valore. Il tipo viene memorizzato e non viene più richiesto alle successive riaperture del report.

L'applicazione mantiene anche (per singolo utente) una cache dei valori inseriti dall'utente in modo da facilitarne il riutilizzo. I valori memorizzati sono accessibili cliccando sul pulsante MRU (Most Recently Used) nella riga del parametro:

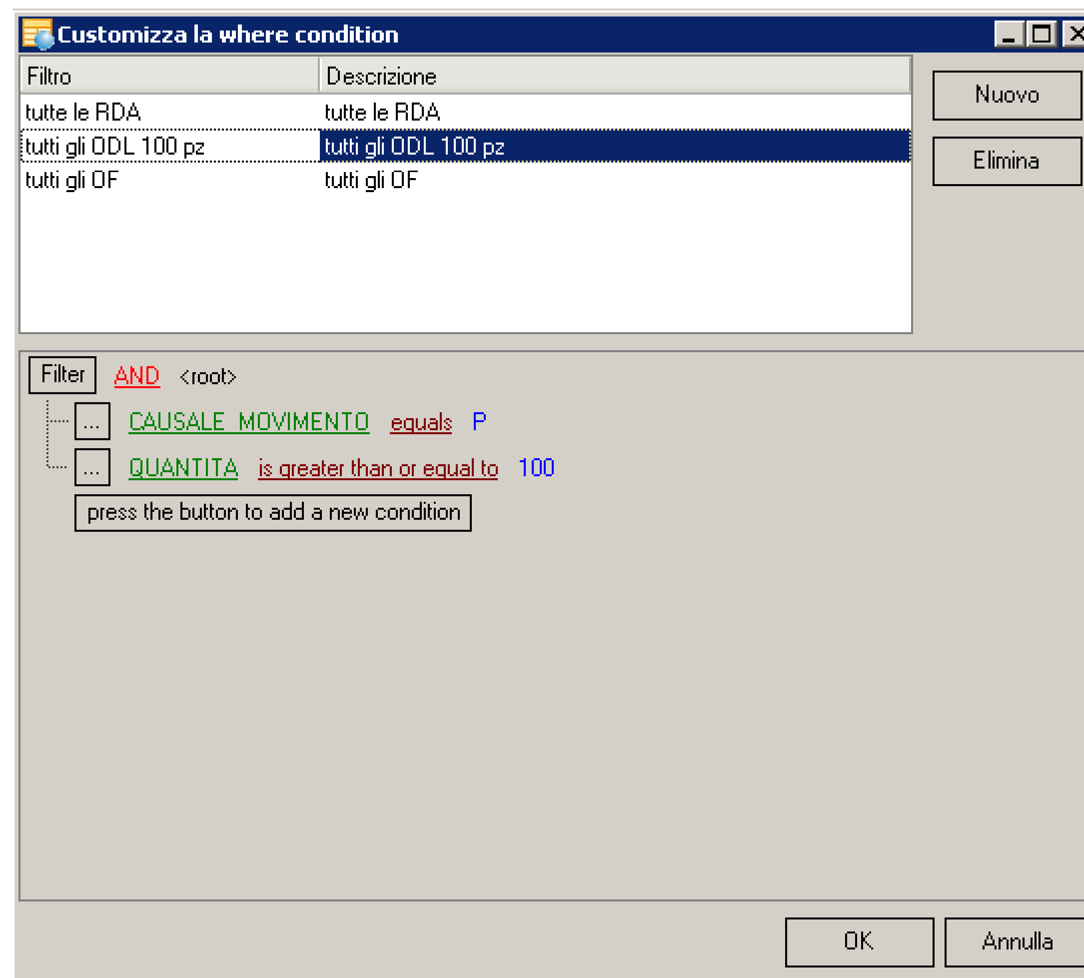


Durante la scrittura della query è possibile aggiungere anche nelle condizioni della SELECT una **condizione predefinita**.
La condizione predefinita viene prelevata da un set di sub-condizioni precedentemente predisposte.

Si procede inserendo nella where condition un tag speciale cliccando sul pulsante "Condizione predefinita". Il tag può essere inserito nello script SQL secondo le necessità dell'utente.



All'apertura di un report che utilizza la query con la condizione customizzabile, viene visualizzata una finestra di dialogo in cui è possibile selezionare la condizione da utilizzare per risolvere la query e dove è possibile creare nuove condizioni. Le condizioni custom vengono costruite come se fossero dei filtri, sarà poi l'applicazione a preoccuparsi di convertirle in SQL quando necessario.



Cliccando sul pulsante "Annulla" la condizione custom non viene valorizzata e viene sostituita da una condizione che sia sempre TRUE: ad esempio in Oracle viene utilizzata la stringa "(1=1)".

6.3.2 Detail Query Manager

Il detail query manager permette di visualizzare le configurazioni master-detail attualmente disponibili.

Quando invocato dalla barra del menu principale alcune funzionalità, che sono disponibili all'atto della creazione di un nuovo report master-detail, sono disabilitate. (Nota tecnica: sono disabilitate tutte le funzionalità per cui è necessario che la maschera conosca la struttura del dataset del report master).

In particolare è possibile:
- modificare la query di una configurazione master-detail già presente
- eliminare una configurazione master-detail già presente
- eliminare una o più condizioni da una configurazione master-detail già presente

Invece non è possibile:
- creare una nuova configurazione master-detail
- modificare una condizione di una configurazione master-detail già presente
- aggiungere una condizione ad una configurazione master-detail già presente

6.3.3 Configura tag per condizione predefinita in query

Le condizioni predefinite sono dei tag `@sample_tag..@` che, annegati nello script SQL di una query master, vengono sostituiti all'apertura del report da una stringa SQL confezionata secondo svariati metodi, tutti configurabili.
La stringa SQL da sostituire al tag può essere generata automaticamente (ad esempio leggendo ed elaborando un set di valori da un file) oppure a fronte di un input fornito dall'utente all'apertura del report (ad esempio facendo scegliere all'utente un valore da un elenco precedentemente predisposto).

Configurazione tag per condizione predefinita in query

Aggiungi Elimina

Nome	Tipo	Etichetta	Priorità
@DURATA@	valuesFromList		0
@ELENCO@	valuesFromList	Elencone	0
@ELENCO_DA_CDB@	valuesFromCDBTable		0
@ELENCO_DA_FILE@	valuesFromTextFile		0
@ELENCO_DA_GERARCHIA	valuesFromHierarchy		0
@ELENCO_DA_QUERY@	valuesFromQuery		0
@FABRIFILTRO@	valuesFromFilter		0
@FILE@	valuesFromTextFile		0
@FILTRONE@	valuesFromFilter		0
@INTERO@	valuesFromParameter		0
@MEMTABLE@	valuesFromCDBTable		0
@PARAMETER@	valuesFromParameter		0
@PARAMETER_ORDER@	valuesFromParameter		0
@PROVA@	valuesFromTextFile		0
@QUERY@	valuesFromQuery		0
@TREE@	valuesFromHierarchy		-5

Tipo del parametro: ftDateTime

Stringa SQL per non caricare alcun dato: TO_DATE('01-01-2000','DD-MM-YYYY')

Stringa SQL da usare se l'utente non sceglie nulla: TO_DATE('01-01-2000','DD-MM-YYYY')

Le priorità serviranno per determinare l'ordine di visualizzazione dei tag (prima quelli con priorità più alta)

OK Annulla

All'interno della stessa query master è possibile inserire uno o più tag.

All'apertura del report di analisi l'utente potrà selezionare i valori/condizioni atti a valorizzare i tag all'interno di un'unica finestra di

dialogo.

Per ogni tag è possibile definire una **priorità** che verrà utilizzata per determinare l'ordine di visualizzazione dei tag quando verranno presentati all'utente per la valorizzazione durante l'apertura dei report. I tag con priorità più alta verranno visualizzati prima di quelli con priorità minore.

6.3.3.1 Tag elenco di valori fissi

Al tag viene sostituito un valore o un elenco di valori estratti da un elenco precedentemente configurato.
Questo tag può essere utilizzato per:

- inserire un elenco di valori prelevati da un set di possibili valori (ad esempio un elenco degli anni o un elenco di nazioni). Qui un esempio.
- iniettare nella query una stringa SQL custom sulla base di una scelta dell'utente. Qui un esempio.

Opzioni di configurazione

Elenco dei valori :
E' possibile inserire uno o più valori arricchendoli da una descrizione (opzionale) che verrà presentata all'utente al posto del valore.

aggiungi apici ai valori :
aggiunge apici ai singoli valori dell'elenco quando viene generata la stringa SQL da sostituire al tag. Nell'esempio iniziale questo valore è stato impostato a true.

racchiudi l'elenco dei valori tra parantesi :
quando viene generata la stringa SQL da sostituire al tag l'elenco dei valori viene racchiusa tra parentesi. Nell'esempio iniziale questo

valore è impostato a true.

Permetti all'utente di

- **scegliere un solo valore** :

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionarle uno attraverso l'apposita finestra.

- **scegliere uno o più valori**:

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionare uno o più valori attraverso l'apposita finestra.

- **non scegliere nulla, tutti i valori sono utilizzati** :

non viene chiesto nulla all'utente, tutti i valori estratti dalla query vengono utilizzati per generare la stringa SQL da sostituire al tag.

Permetti all'utente di inserire testo libero :

l'utente può modificare a mano la stringa SQL generata a fronte dei valori che ha selezionato. L'utente può anche inserire una stringa completamente riscritta da zero.

Stringa SQL da usare come valore nullo :

questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.

Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore.

Ad esempio, se il tag rappresenta un insieme di valori racchiusi tra parentesi e con apici, una stringa valida come valore nullo potrebbe essere: `('PIPPO')`.

Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

6.3.3.1.1 Esempio tag elenco di valori fissi

Creiamo una query master: `SELECT * FROM ALLOCAZIONI WHERE CODICE_RISORSA IN @ELENCO@`

Configuriamo il tag `@ELENCO@` in questo modo:

Configurazione tag per condizione predefinita in query

Aggiungi Elimina

Nome	Tipo	Etichetta
@ELENCO@	valuesFromList	Elencone
@ELENCO_DA_CDB@	valuesFromCDBTable	
@ELENCO_DA_FILE@	valuesFromTextFile	
@ELENCO_DA_GERARCHIA	valuesFromHierarchy	
@ELENCO_DA_QUERY@	valuesFromQuery	
@FILE@	valuesFromTextFile	
@FILTRONE@	valuesFromFilter	
@MEMTABLE@	valuesFromCDBTable	
@PARAMETER@	valuesFromParameter	
@PROVA@	valuesFromTextFile	
@QUERY@	valuesFromQuery	
@TREE@	valuesFromHierarchy	

Valore	Descrizione
MISC-FOGLIARI-2	Miscelatore grosso
REATTORE-1	Reattore che non funziona
CONF-1KG	Confezionatrice blu
TR1	Tritatutto

☑ aggiungi apici ai valori
☑ racchiudi l'elenco dei valori tra parentesi

Permetti all'utente di..

○ scegliere un solo valore
◉ scegliere uno o più valori
○ non scegliere nulla, tutti i valori sono utilizzati

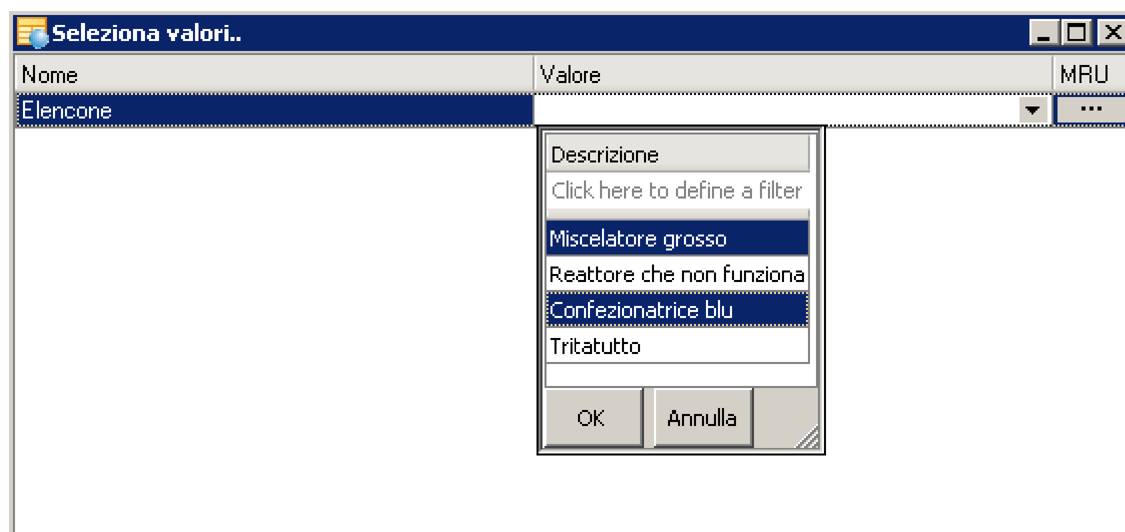
Esempio: ('value1','value2','value3')

☐ Permetti all'utente di inserire testo libero

Stringa SQL da usare come valore nullo:
('--no-value--')

OK Annulla

All'apertura del report l'utente potrà selezionare uno o più valori dall'elenco:



Il tag verrà sostituito da una stringa SQL simile a `('MISC-FOGLIARI-2','CONF-1KG')`

6.3.3.1.2 Esempio tag elenco di valori fissi 2

Creiamo una query master: `SELECT CODICE_ORDINE, NUMERO_FASE, @DURATA@ AS DURATA FROM ALLOCAZIONI`

Configuriamo il tag `@DURATA@` in questo modo:

Configurazione tag per condizione predefinita in query

Aggiungi Elimina

Nome	Tipo	Etichetta
@DURATA@	valuesFromList	
@ELENCO@	valuesFromList	Elencone
@ELENCO_DA_CDB@	valuesFromCDBTable	
@ELENCO_DA_FILE@	valuesFromTextFile	
@ELENCO_DA_GERARCHIA(	valuesFromHierarchy	
@ELENCO_DA_QUERY@	valuesFromQuery	
@FILE@	valuesFromTextFile	
@FILTRONE@	valuesFromFilter	
@MEMTABLE@	valuesFromCDBTable	
@PARAMETER@	valuesFromParameter	
@PROVA@	valuesFromTextFile	
@QUERY@	valuesFromQuery	
@TREE@	valuesFromHierarchy	

Valore	Descrizione
(tempo_prep_simulato + tempo_avv_simulato + tempo_lav_simulato)	durata totale
tempo_lav_simulato	solo lavorazione

☐ aggiungi apici ai valori

☐ racchiudi l'elenco dei valori tra parentesi

Permetti all'utente di...

◉ scegliere un solo valore

○ scegliere uno o più valori

○ non scegliere nulla, tutti i valori sono utilizzati

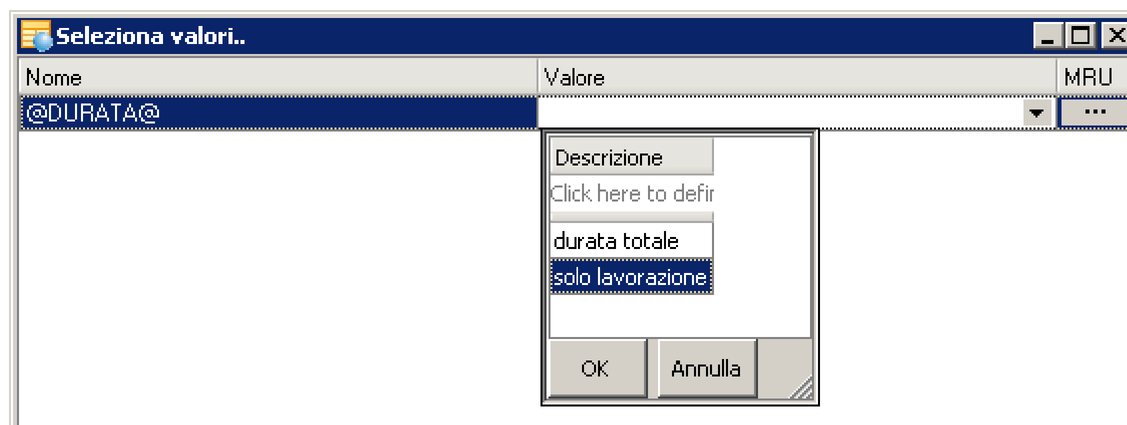
☐ Permetti all'utente di inserire testo libero

Stringa SQL da usare come valore nullo:

tempo_lav_simulato

OK Annulla

All'apertura del report l'utente potrà selezionare uno o più valori dall'elenco:



Il tag verrà sostituito da una stringa SQL tra quelle inserite in precedenza.

Ecco come cambia il report operando scelte differenti alla partenza:

Report di analisi

QUERY_TAG

Scenario | Filtro

Drag a column header here to group by that column

Codice ordine	Numero	Durata
200600047050	10,00	,05
200600047047	10,00	41,51
200600047042	10,00	8,53
200600047041	10,00	19,09
200600047038	10,00	8,53
200600047038	20,00	6,00
200600047033	10,00	3,20
200600047031	10,00	4,27
200600047030	10,00	2,67

Report di analisi

QUERY_TAG

Scenario | Filtro

Drag a column header here to group by that column

Codice ordine	Numero	Durata
200600047050	10,00	,05
200600047047	10,00	41,51
200600047042	10,00	9,03
200600047041	10,00	19,09
200600047038	10,00	10,53
200600047038	20,00	10,00
200600047033	10,00	4,70
200600047031	10,00	4,27
200600047030	10,00	4,17

6.3.3.2 Tag elenco di valori da query

Al tag viene sostituito un valore o un elenco di valori estratti dal database attraverso una query. Qui un esempio.

Opzioni di configurazione

Campo chiave :

Il nome del campo della query da utilizzare per generare l'elenco dei valori.

Sql :
La query con cui estrarre i valori da presentare all'utente per la scelta.

aggiungi apici ai valori :
aggiunge apici ai singoli valori dell'elenco quando viene generata la stringa SQL da sostituire al tag. Nell'esempio iniziale questo valore è stato impostato a true.

racchiudi l'elenco dei valori tra parantesi :
quando viene generata la stringa SQL da sostituire al tag l'elenco dei valori viene racchiusa tra parentesi. Nell'esempio iniziale questo valore è impostato a true.

Permetti all'utente di

- **scegliere un solo valore** :

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionarle uno attraverso l'apposita finestra.

- **scegliere uno o più valori**:

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionare uno o più valori attraverso l'apposita finestra.

- **non scegliere nulla, tutti i valori sono utilizzati** :

non viene chiesto nulla all'utente, tutti i valori estratti dalla query vengono utilizzati per generare la stringa SQL da sostituire al tag.

Permetti all'utente di inserire testo libero :
l'utente può modificare a mano la stringa SQL generata a fronte dei valori che ha selezionato. L'utente può anche inserire una stringa completamente riscritta da zero.

Stringa SQL da usare come valore nullo :

questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.

Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore.

Ad esempio, se il tag rappresenta un insieme di valori racchiusi tra parentesi e con apici, una stringa valida come valore nullo potrebbe essere: `('PIPPO')`

Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

6.3.3.2.1 Esempio tag elenco di valori da query

Creiamo una query master: `SELECT * FROM ALLOCAZIONI WHERE CODICE_RISORSA IN @ELENCO_DA_QUERY@`

Colleghiamo al tag la query `SELECT CODICE_MACCHINA, DESCRIZIONE FROM MACCHINE` e impostiamo come campo chiave `CODICE_MACCHINA` e consentiamo all'utente di scegliere uno o più valori

Configurazione tag per condizione predefinita in query

Aggiungi Elimina

Nome	Tipo	Etichetta
@ELENCO@	valuesFromList	Elencone
@ELENCO_DA_FILE@	valuesFromTextFile	
@FILE@	valuesFromTextFile	
@FILTRONE@	valuesFromFilter	
@MEMTABLE@	valuesFromCDBTable	
@PARAMETER@	valuesFromParameter	
@PROVA@	valuesFromTextFile	
@QUERY@	valuesFromQuery	
@TREE@	valuesFromHierarchy	
@ELENCO_DA_QUERY@	valuesFromQuery	

Campo chiave:

CODICE_MACCHINA

```
SELECT CODICE_MACCHINA, DESCRIZIONE FROM
MACCHINE
```

☐ a capo automatico

☑ aggiungi apici ai valori

☑ racchiudi l'elenco dei valori tra parentesi

Permetti all'utente di..

○ scegliere un solo valore

◉ scegliere uno o più valori

○ non scegliere nulla, tutti i valori sono utilizzati

Esempio: ('value1','value2','value3')

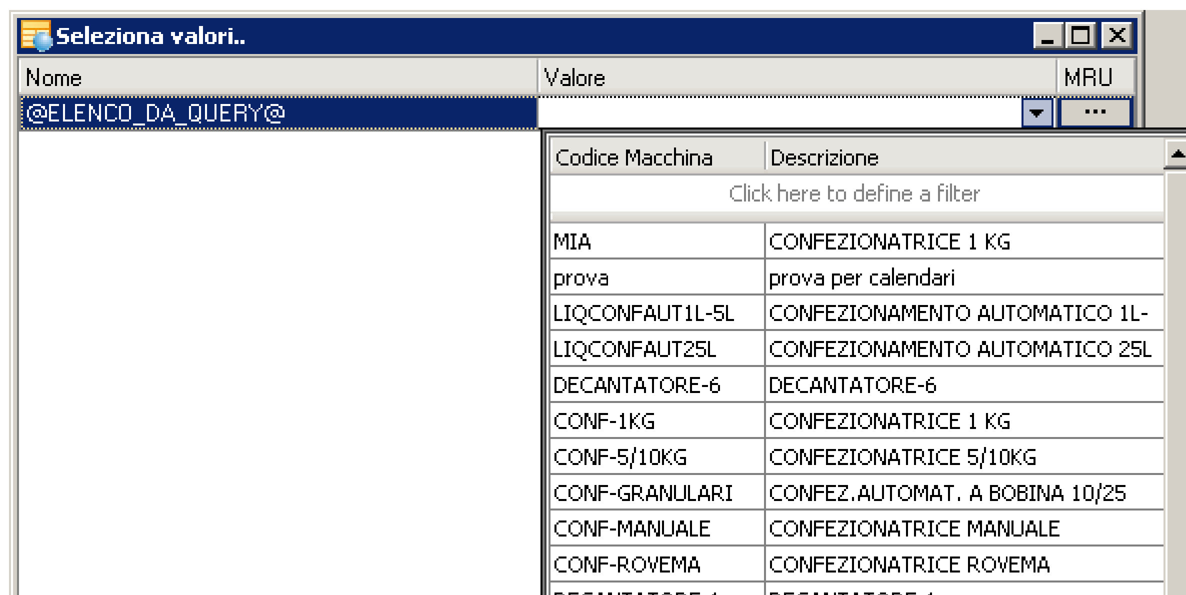
☐ Permetti all'utente di inserire testo libero

Stringa SQL da usare come valore nullo:

('--no-value--')

OK Annulla

All'apertura del report l'utente può selezionare dall'apposita tendina una o più risorse:



Il tag verrà sostituito da una stringa SQL simile a `('RISORSA1', 'RISORSA2', 'RISORSA3')`

6.3.3.3 Tag elenco di valori da tabella del CDB

Al tag viene sostituito un valore o un elenco di valori estratti dal database attraverso una query. Qui un esempio.

Opzioni di configurazione

Tabella :

La tabella del db in memoria (file CDB) con cui estrarre i valori da presentare all'utente per la scelta.

Campo chiave :

Il nome del campo della tabella da utilizzare per generare l'elenco dei valori.

aggiungi apici ai valori :

aggiunge apici ai singoli valori dell'elenco quando viene generata la stringa SQL da sostituire al tag. Nell'esempio iniziale questo valore è stato impostato a true.

racchiudi l'elenco dei valori tra parantesi :

quando viene generata la stringa SQL da sostituire al tag l'elenco dei valori viene racchiusa tra parentesi. Nell'esempio iniziale questo valore è impostato a true.

Permetti all'utente di

- **scegliere un solo valore** :

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionarle uno attraverso l'apposita finestra.

- **scegliere uno o più valori**:

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionare uno o più valori attraverso l'apposita finestra.

- **non scegliere nulla, tutti i valori sono utilizzati** :

non viene chiesto nulla all'utente, tutti i valori estratti dalla query vengono utilizzati per generare la stringa SQL da sostituire al tag.

Permetti all'utente di inserire testo libero :

l'utente può modificare a mano la stringa SQL generata a fronte dei valori che ha selezionato. L'utente può anche inserire una stringa completamente riscritta da zero.

Stringa SQL da usare come valore nullo :

questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.

Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore.

Ad esempio, se il tag rappresenta un insieme di valori racchiusi tra parentesi e con apici, una stringa valida come valore nullo potrebbe essere: `('PIPPO')`

Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

6.3.3.3.1 Esempio tag elenco di valori da tabella

Creiamo una query master: `SELECT * FROM ALLOCAZIONI WHERE CODICE_RISORSA IN @ELENCO_DA_CDB@`

Colleghiamo al tag la tabella in memoria `RISORSE` e impostiamo come campo chiave `CODICE_RISORSA` e consentiamo all'utente di scegliere uno o più valori

Configurazione tag per condizione predefinita in query

Aggiungi | Elimina

Nome	Tipo	Etichetta
@ELENCO@	valuesFromList	Elencone
@ELENCO_DA_CDB@	valuesFromCDBTable	
@ELENCO_DA_FILE@	valuesFromTextFile	
@ELENCO_DA_QUERY@	valuesFromQuery	
@FILE@	valuesFromTextFile	
@FILTRONE@	valuesFromFilter	
@MEMTABLE@	valuesFromCDBTable	
@PARAMETER@	valuesFromParameter	
@PROVA@	valuesFromTextFile	
@QUERY@	valuesFromQuery	
@TREE@	valuesFromHierarchy	

Tabella: RISORSE

Campo chiave: CODICE_RISORSA

- [x] aggiungi apici ai valori
- [x] racchiudi l'elenco dei valori tra parentesi

Permetti all'utente di..
- ( ) scegliere un solo valore
- (•) scegliere uno o più valori
- ( ) non scegliere nulla, tutti i valori sono utilizzati

Esempio: ('value1','value2','value3')

- [ ] Permetti all'utente di inserire testo libero

Stringa SQL da usare come valore nullo: ('--no-value--')

OK | Annulla

All'apertura del report l'utente può selezionare dall'apposita tendina una o più risorse:

Seleziona valori..

Nome	Valore	MRU
@ELENCO_DA_CDB@		...

Codice risorsa	Tipo risorsa	Sel risorsa	Descrizione	Parte di	Valenza	Sottovalenza	Turno base	Capacita finita	Entita terminale	Calendario ereditato d	Internc	Data ultimo
Click here to define a filter												
CLIQ5	O	1	CONFEZIONAMENTO LIQUIDI 5 LIT		1	1	BASE	1	1	CLIQ5	1	
31789	1	1	SERBATOIO BULK 31789		1	1		1	0		1	
V31789	3	1	VERSAMENTO IN 31789		1	1		1	0		1	
P31789	3	1	PRELIEVO DA 31789		1	1		1	0		1	
31790	1	1	SERBATOIO BULK 31790		1	1		1	0		1	
31791	1	1	SERBATOIO BULK 31791		1	1		1	0		1	
P31790	3	1	PRELIEVO DA 31790		1	1		1	0		1	
P31791	3	1	PRELIEVO DA 31791		1	1		1	0		1	
V31790	3	1	VERSAMENTO IN 31790		1	1		1	0		1	
V31791	3	1	VERSAMENTO IN 31791		1	1		1	0		1	
LIQCONFAUT1L-5L	M	1	CONFEZIONAMENTO AUTOMATICO 1L-	CDL03	1	1	BASE	1	1	LIQCONFAUT1L-5L	1	11/06/201C
LIQCONFAUT25L	M	1	CONFEZIONAMENTO AUTOMATICO 25L	CDL03	1	1	BASE	1	1	LIQCONFAUT25L	1	11/06/201C

Il tag verrà sostituito da una stringa SQL simile a `('RISORSA1', 'RISORSA2', 'RISORSA3')`

6.3.3.4 Tag elenco di valori da file

Al tag viene sostituito un valore o un elenco di valori estratti da un file di testo. Qui un esempio.

Opzioni di configurazione

File : nome del file da cui estrarre i valori. Può essere inserito il nome di un file completo di tutto il percorso (ad esempio `C:\ELENCO_RISORSE.TXT`) oppure, in abbinamento con l'opzione seguente, un percorso relativo che viene risolto usando le cartelle di ACN del company chart come prefisso (ad esempio `SOTTOCARTELLA\ELENCO_RISORSE.TXT` che verrà cercato in `[ACN]\PROFILES\[UTENTE]`

\ATOMOSVISION\SOTTOCARTELLA\ELENCO_RISORSE.TXT e, se non trovato, nelle corrispondenti sottocartelle dei livelli precedenti nel company chart)

cerca il file indicato nelle cartelle di ACN :
Se abilitato, cerca il file nelle sottocartelle di ACN. Vedi sopra.

aggiungi apici ai valori :
aggiunge apici ai singoli valori dell'elenco quando viene generata la stringa SQL da sostituire al tag. Nell'esempio iniziale questo valore è stato impostato a true.

racchiudi l'elenco dei valori tra parantesi :
quando viene generata la stringa SQL da sostituire al tag l'elenco dei valori viene racchiusa tra parentesi. Nell'esempio iniziale questo valore è impostato a true.

Permetti all'utente di

- **scegliere un solo valore** :

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionarle uno attraverso l'apposita finestra.

- **scegliere uno o più valori**:

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionare uno o più valori attraverso l'apposita finestra.

- **non scegliere nulla, tutti i valori sono utilizzati** :

non viene chiesto nulla all'utente, tutti i valori presenti nel file vengono utilizzati per generare la stringa SQL da sostituire al tag.

Permetti all'utente di inserire testo libero :
l'utente può modificare a mano la stringa SQL generata a fronte dei valori che ha selezionato. L'utente può anche inserire una stringa

completamente riscritta da zero.

Stringa SQL da usare come valore nullo :
questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.
Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore.
Ad esempio, se il tag rappresenta un insieme di valori racchiusi tra parentesi e con apici, una stringa valida come valore nullo potrebbe essere: `('PIPPO')`
Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

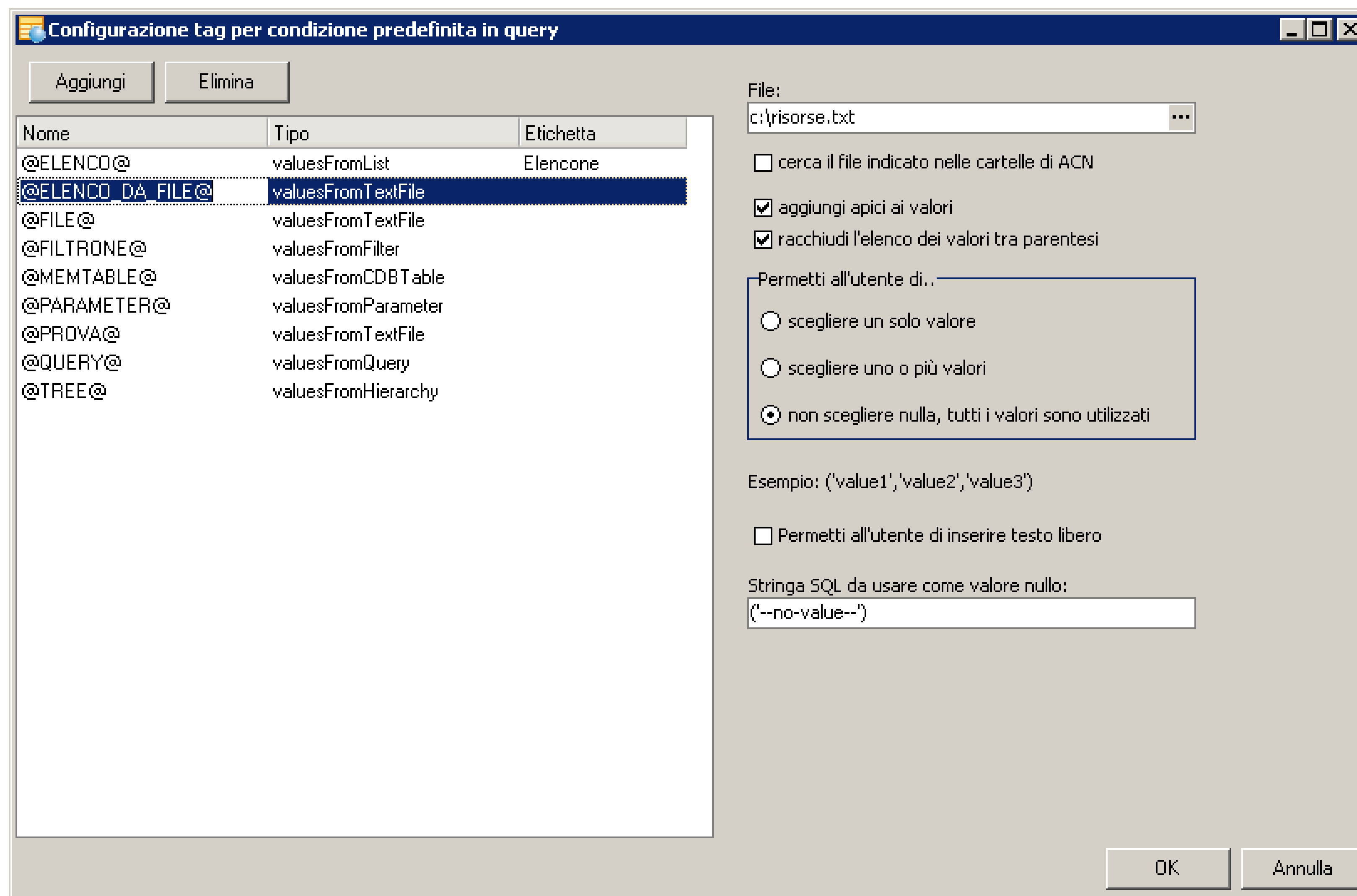
6.3.3.4.1 Esempio tag elenco di valori da file

Creiamo una query master: `SELECT * FROM ALLOCAZIONI WHERE CODICE_RISORSA IN @ELENCO_DA_FILE@`

Creiamo un file `C:\RISORSE.TXT` che contiene 3 righe:

```
RISORSA1
RISORSA2
RISORSA3
```

Impostiamo la scelta automatica di tutti i valori (senza permette all'utente di selezionare alcunché):



All'apertura del report il tag viene sostituito da questa stringa SQL: `('RISORSA1', 'RISORSA2', 'RISORSA3')`

per cui la query che alimenterà il report diventerà questa:

```
SELECT * FROM ALLOCAZIONI WHERE CODICE_RISORSA IN ('RISORSA1', 'RISORSA2', 'RISORSA3')
```

6.3.3.5 Tag elenco di valori da gerarchia

Al tag viene sostituita una stringa SQL che contiene una o più sequenze di condizioni in AND messe in OR tra di loro. Qui un esempio.

Opzioni di configurazione

Campi chiave (separati da ;) :

E' l'elenco dei campi della query SQL utilizzata per costruire la gerarchia che devono essere coinvolti nella generazione della stringa SQL da sostituire al tag.

SQL :

E' la query a partire dalla quale verrà costruita la gerarchia. La gerarchia è costruita come un group by su uno o più campi.

Configura gerarchia :

L'elenco dei campi della query della gerarchia che bisogna raggruppare.

Etichette dei campi :

E' possibile definire degli alias dei campi della query della gerarchia in modo che vengano utilizzati durante la generazione della stringa SQL da sostituire al tag.

Mantieni una cache della gerarchia :

Mantieni in memoria la gerarchia dopo la prima apertura del report in modo da velocizzare le successive aperture.

aggiungi apici ai valori :

aggiunge apici ai singoli valori dell'elenco quando viene generata la stringa SQL da sostituire al tag. Nell'esempio iniziale questo valore è stato impostato a true.

Nodi utilizzati:

- usa i nodi di tutti i livelli :

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionarle uno attraverso l'apposita finestra.

- usa solo i nodi terminali:

viene letto l'elenco di valori dal file e viene chiesto all'utente di selezionare uno o più valori attraverso l'apposita finestra.

Permetti all'utente di inserire testo libero :

l'utente può modificare a mano la stringa SQL generata a fronte dei valori che ha selezionato. L'utente può anche inserire una stringa completamente riscritta da zero.

Stringa SQL da usare come valore nullo :

questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.

Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore.

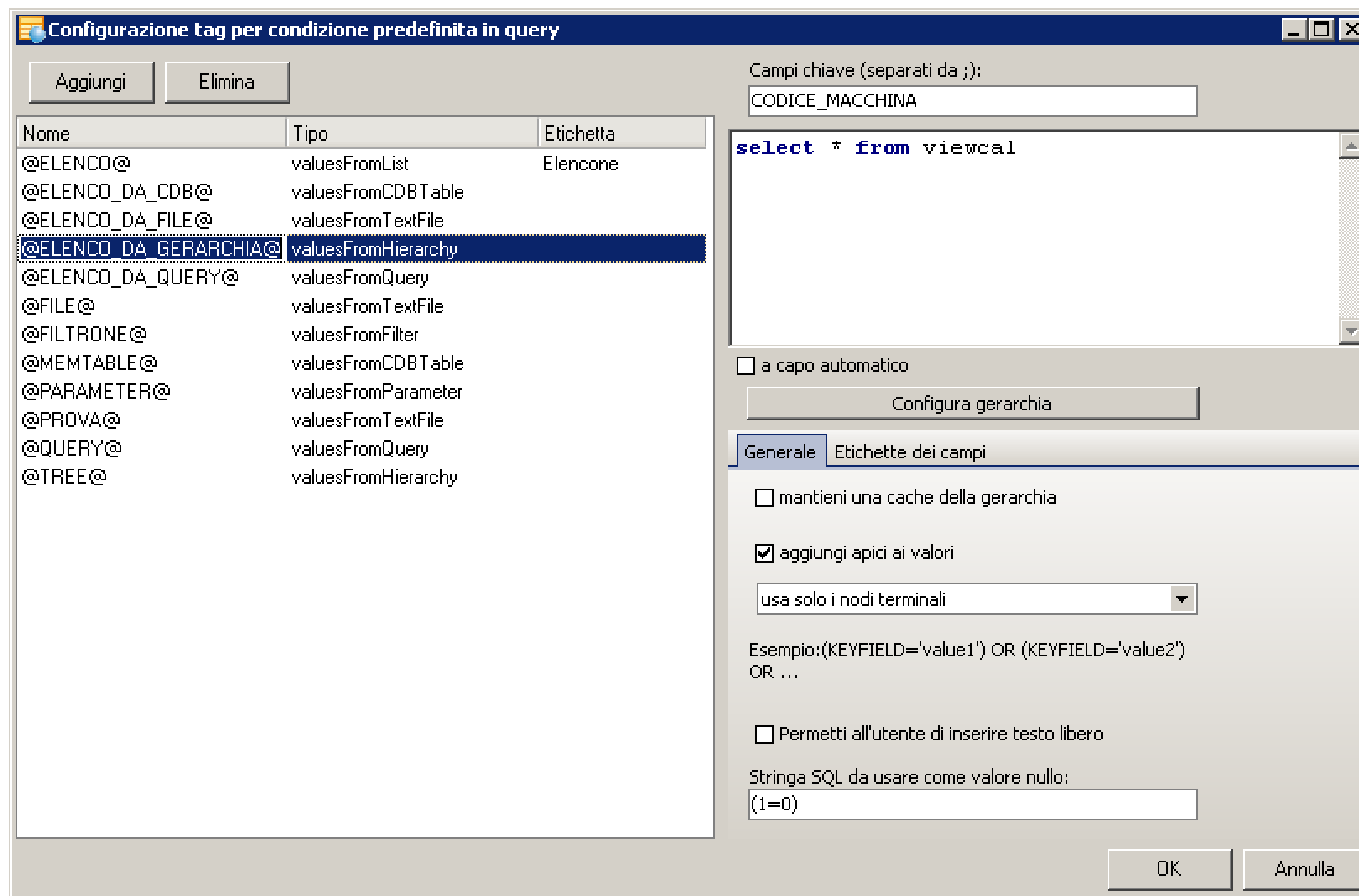
Ad esempio, se il tag rappresenta una gerarchia una stringa valida come valore nullo potrebbe essere: `(1=0)`

Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

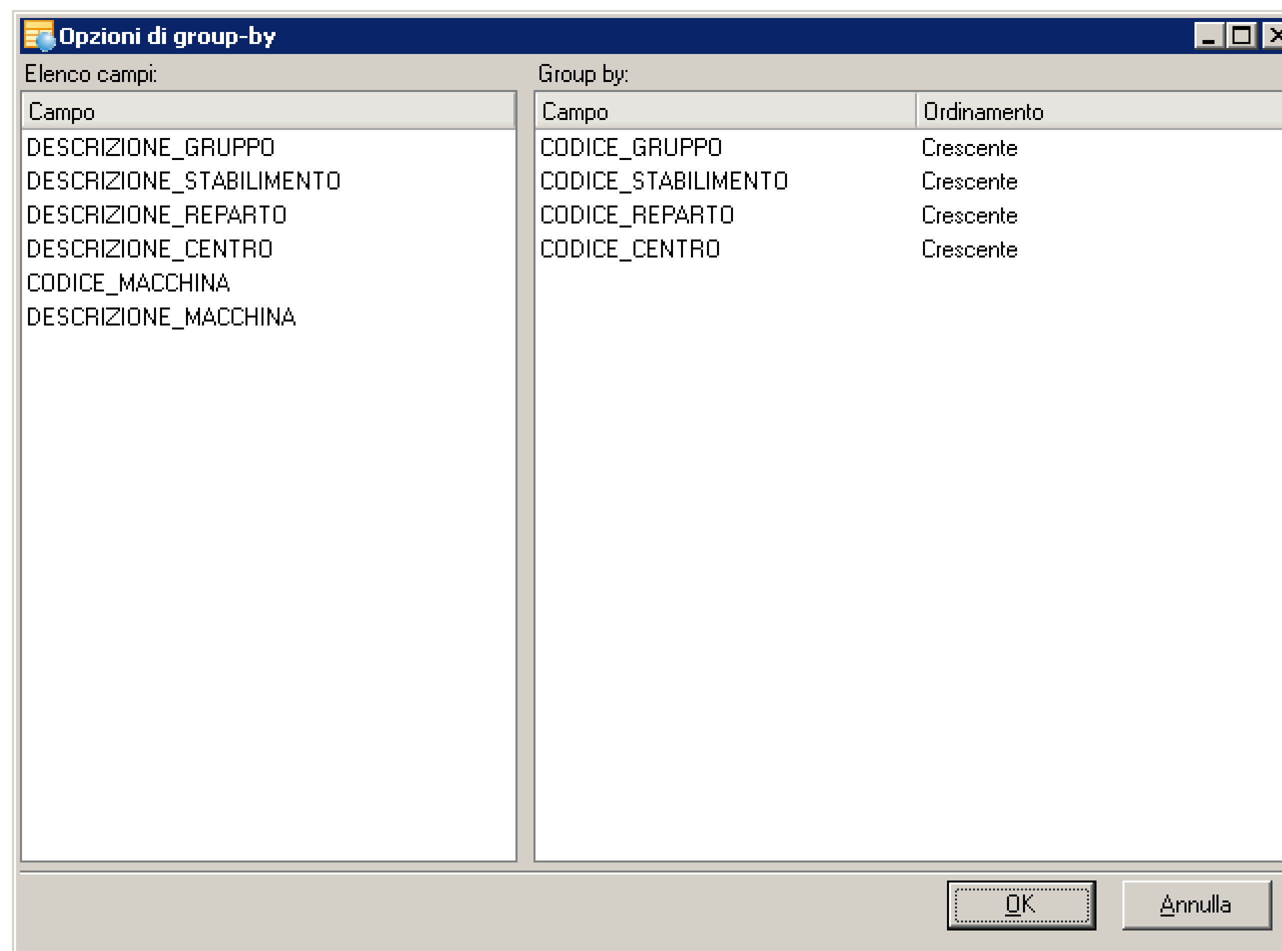
6.3.3.5.1 Esempio tag elenco di valori da gerarchia

Creiamo una query master: `SELECT * FROM ALLOCAZIONI WHERE @ELENCO_DA_GERARCHIA@`

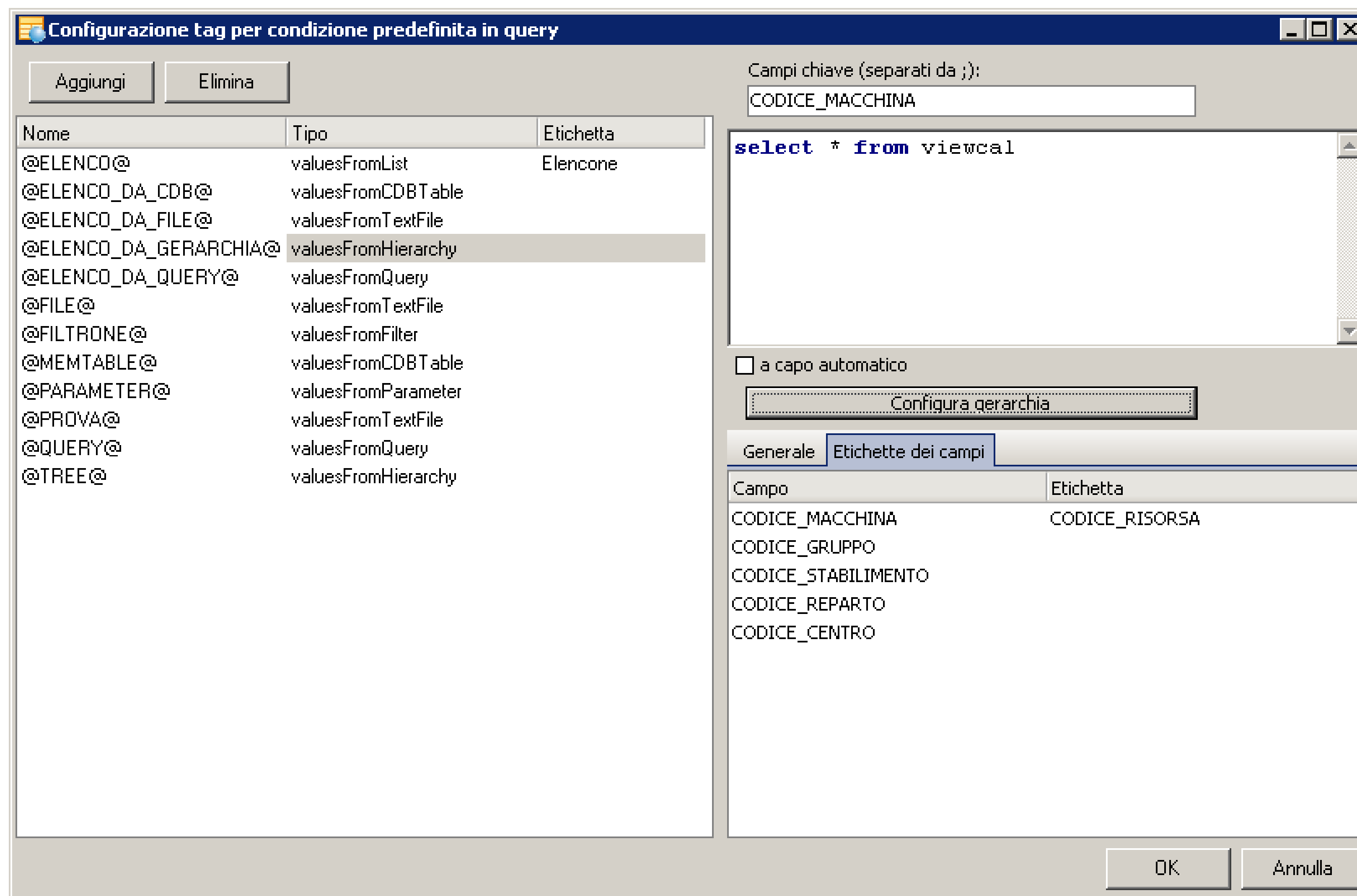
Inseriamo come query per generare la gerarchia `SELECT * FROM VIEWCAL`, impostiamo `CODICE_MACCHINA` come campo chiave e l'utilizzo dei soli nodi terminali nella generazione dell'SQL da sostituire al tag.



La gerarchia viene configurata in questo modo:



E' anche necessario cambiare la label del campo `CODICE_MACCHINA` in `CODICE_RISORSA`:



All'apertura del report l'utente può selezionare dall'apposita tendina uno o più nodi della gerarchia. Selezionando un nodo vengono

selezionati anche tutti i nodi figli ed i padri appartenenti al percorso minimo per raggiungere la radice (nell'esempio si è cliccato sulla risorsa CDL02):

Seleziona valori..

Nome	Valore	MRU
@ELENCO_DA_GERARCHIA@		...

		Codice Gruppo	Descrizione Gruppo	Codice Stabilimento	Descrizione Stabilimento	Codice Reparto	Descrizione Reparto	Codice Cent
Valagro	☑							
ATESSA	☑							
REP01	☑							
CDL01	☐							
prova	☐	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL01
ESTRATTORE	☐	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL01
CDL02	☑							
REATTORE-1	☑	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL02
REATTORE-2	☑	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL02
REATTORE-3	☑	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL02
REATTORE-4	☑	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL02
REATTORE-5	☑	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL02
CDL03	☐							
LIQCONFAUT1L-5L	☐	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL03
LIQCONFAUT25L	☐	Valagro	Gruppo Valagro	ATESSA	Stabilimento prod. di Atessa	REP01	LIQUIDI	CDL03

La stringa SQL generata è : `(CODICE_RISORSA='REATTORE-1') OR (CODICE_RISORSA='REATTORE-2') OR (CODICE_RISORSA='REATTORE-3') OR (CODICE_RISORSA='REATTORE-4') OR (CODICE_RISORSA='REATTORE-5')`

6.3.3.6 Tag simil-parametro

Al tag viene sostituito un valore inserito dall'utente. Qui un esempio.

Opzioni di configurazione

Tipo del parametro :

A scelta tra

- string
- integer
- float
- date/time

Stringa SQL da usare come valore nullo :

questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.

Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore.

Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

6.3.3.6.1 Esempio tag simil-parametro

Creiamo una query master: `SELECT * FROM ALLOCAZIONI WHERE DATA_ORA_INIZIO_SIMULATA >  @PARAMETER@`

Configuriamo il tag `@PARAMETER@` in questo modo:

Configurazione tag per condizione predefinita in query

Aggiungi Elimina

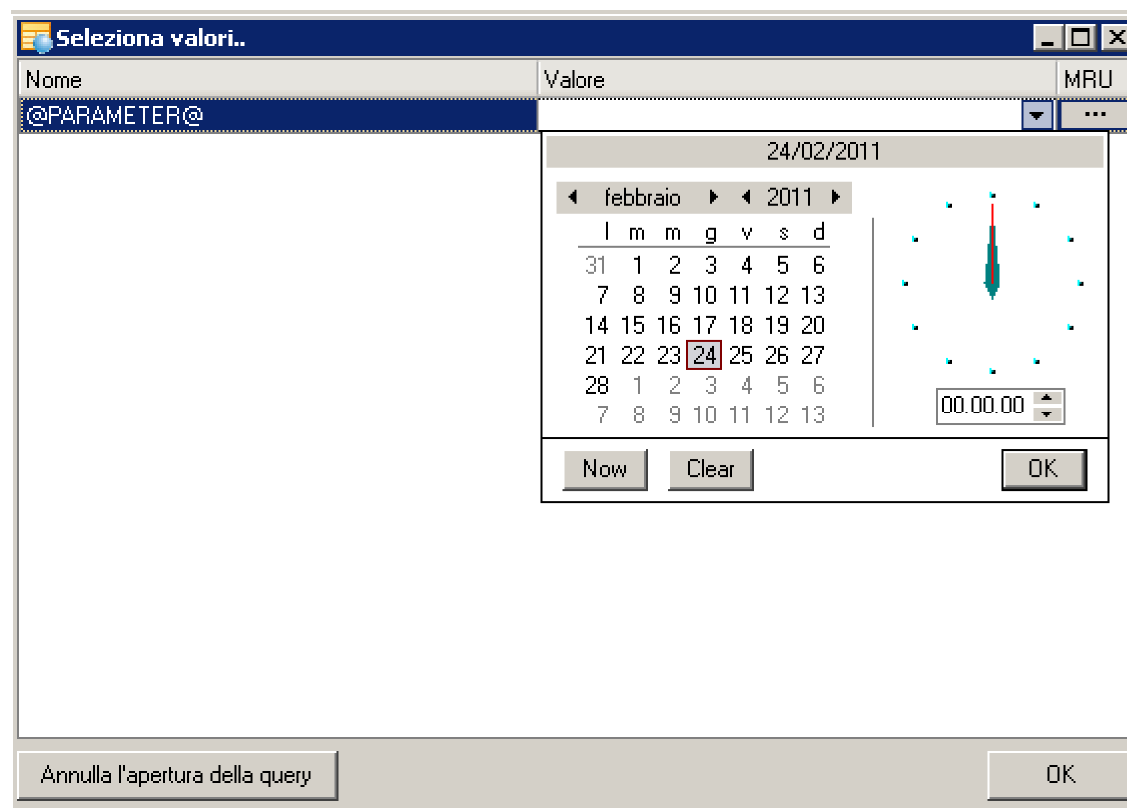
Nome	Tipo	Etichetta
@ELENCO@	valuesFromList	Elencone
@ELENCO_DA_CDB@	valuesFromCDBTable	
@ELENCO_DA_FILE@	valuesFromTextFile	
@ELENCO_DA_GERARCHIA	valuesFromHierarchy	
@ELENCO_DA_QUERY@	valuesFromQuery	
@FILE@	valuesFromTextFile	
@FILTRONE@	valuesFromFilter	
@MEMTABLE@	valuesFromCDBTable	
@PARAMETER@	valuesFromParameter	
@PROVA@	valuesFromTextFile	
@QUERY@	valuesFromQuery	
@TREE@	valuesFromHierarchy	

Tipo del parametro:
ftDateTime

Stringa SQL da usare come valore nullo:
TO_DATE('01-01-2000','DD-MM-YYYY')

OK Annulla

All'apertura del report, questa maschera verrà proposta all'utente:



6.3.3.7 Tag simil-filtro

Al tag viene sostituito una stringa SQL generata a fronte di un filtro creato dall'utente (al volo o selezionato da un elenco di filtri precedentemente creati). La composizione del filtro è fatta attraverso la stessa maschera dei filtri utilizzata nei vari report, per cui in modo visuale e non è necessario che l'utente conosca la sintassi SQL. Qui un esempio.

Opzioni di configurazione

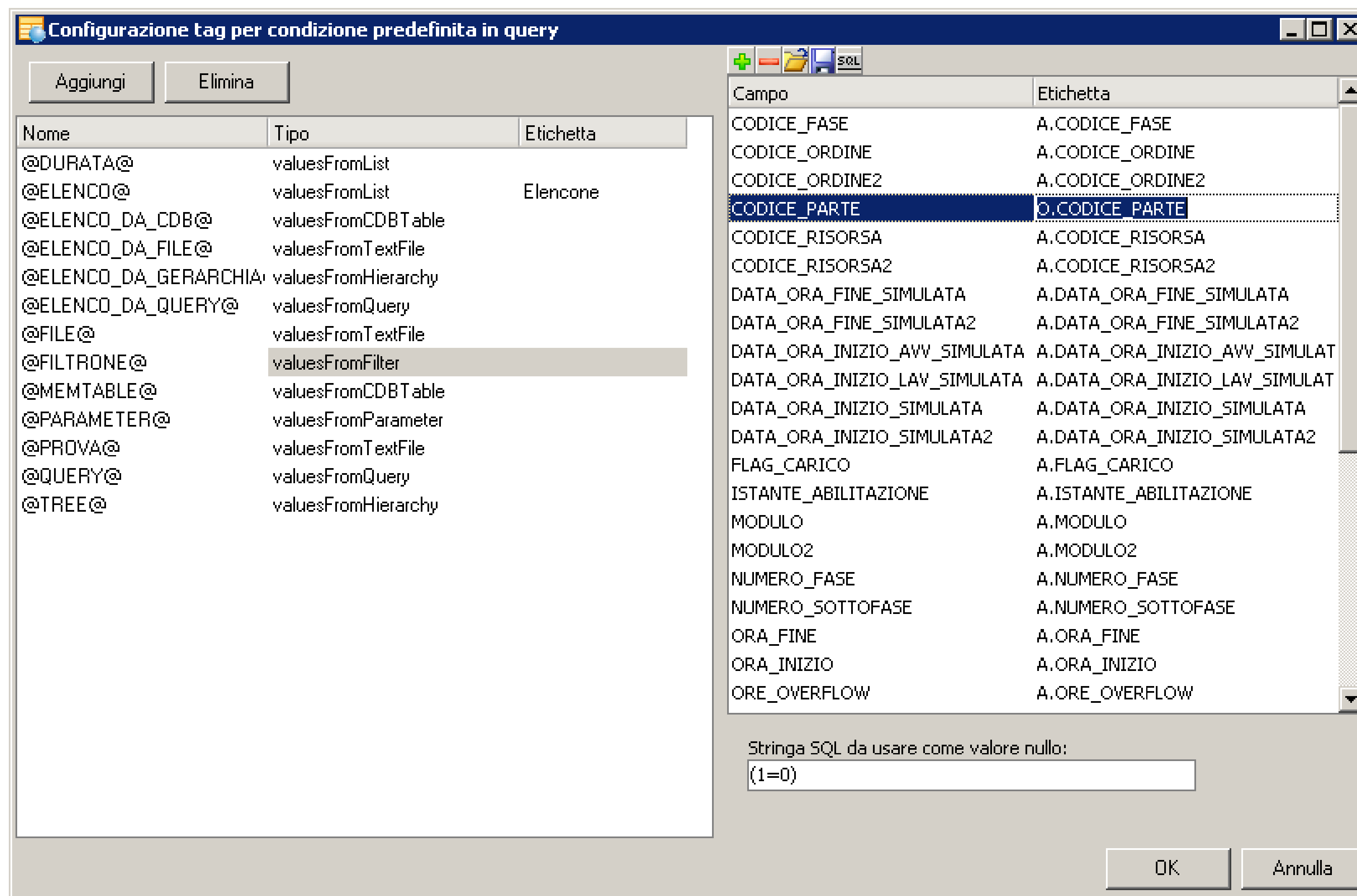
Etichette dei campi :
E' possibile definire degli alias dei campi della query in modo che vengano utilizzati durante la generazione della stringa SQL da sostituire al tag. Sono fondamentali se le query in cui si desidera utilizzare il tag contengono delle join tra due e più tabelle per cui nella where condition è obbligatorio identificare i campi antecedendovi un prefisso (ad esempio A.NOME_CAMPO).
E' possibile caricare le etichette da un file di testo ed anche salvarle su di un file per il riutilizzo.
E' anche possibile inserire al volo una query SQL per estrarre rapidamente un elenco di campi.

Stringa SQL da usare come valore nullo :
questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.
Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura della query ma senza ritornare alcun valore, ad esempio `(1=0)`.
Questo valore è importante anche in caso di più tag inseriti all'interno della stessa query, al riguardo vedere questa sezione.

6.3.3.7.1 Esempio tag simil-filtro

Creiamo una query master: `select a.*, o.codice_parte from allocazioni a , odl o where (a.codice_ordine=o.codice_ordine) and @FILTRONE@`

Configuriamo il tag `@FILTRONE@` in questo modo:



Le etichette dei campi sono state estratte a partire da questa query SQL inserita al volo cliccando sull'ultimo pulsante della toolbar:

```
select a.*, o.codice_parte from allocazioni a , odl o where (a.codice_ordine=o.codice_ordine) and (1=0)
```

All'apertura del report è possibile definire e utilizzare una condizione complessa senza conoscere l'SQL:



Nell'esempio il tag verrà sostituito dalla stringa `(O.CODICE_PARTE LIKE '%1')`

6.3.3.8 Valori nulli per i tag

Per ogni tag è possibile definire una stringa SQL che rappresenta il valore nullo.

Questa stringa è quella che viene utilizzata quando l'utente clicca sul pulsante "Annulla" nella finestra di selezione dei valori dei tag o quando non seleziona alcun valore tra quelli proposti.

Si consiglia di utilizzare una stringa SQL corretta dal punto di vista sintattico e che, se inserita nella query originale, permetta l'apertura

della query ma senza ritornare alcun valore.

Questo valore è importante anche nel caso di più tag inseriti all'interno della medesima query nel caso in cui l'utente decidesse di non valorizzarli tutti.
Questa condizione potrebbe essere esplicitamente prevista da chi crea il report in quanto potrebbe essere utile dare all'utente la possibilità di sezionare i valori che alimentano un report in modi diversi, pur rimanendo sempre con un solo report definito.
Se i dati iniziali fossero degli ordini di lavoro, potrebbe essere definita una modalità di filtraggio per articolo/parte ma anche una alternativa per ordine cliente servito, tutto all'interno della medesimo report/query master utilizzando ad esempio 2 tag diversi il cui valore nullo deve quindi essere opportunamente configurato.

Vediamo un esempio semplice:

Vogliamo dare all'utente la possibilità di filtrare obbligatoriamente le righe della tabella ALLOCAZIONI per risorsa ma anche facoltativamente per data di inizio dell'allocazione.

Definiamo allora una query master in questo modo:

```
select * from allocazioni where (codice_risorsa in  @ELENCO_DA_CDB@) and (data_ora_inizio_simulata >  @PARAMETER@)
```

Il tag `@ELENCO_DA_CDB@` è definito come in questo esempio.
Il tag `@PARAMETER@` è definito come in questo esempio.

All'apertura del report la maschera che viene presentata all'utente è questa:

Seleziona valori..

Nome	Valore	MRU
@ELENCO_DA_CDB@		...
@PARAMETER@		...

Annulla l'apertura della query OK

Questo è l'input inserito dall'utente che ha evidentemente scelto di filtrare le allocazioni di sole 2 risorse ma non desidera mettere alcun limite temporale alla data schedulata:

Seleziona valori..

Nome	Valore	MRU
@ELENCO_DA_CDB@	('LIQCONFAUT1L-5L','LIQCONFAUT25L')	...
@PARAMETER@		...

Annulla l'apertura della query OK

Alla fine questa è la query che andrà ad alimentare il report, il tag `@PARAMETER@` assumerà il valore definito come valore nullo nelle opzioni del tag:

```
select * from allocazioni where (codice_risorsa in  ('LIQCONFAUT1L-5L','LIQCONFAUT25L')) and (data_ora_inizio_simulata >  TO_DATE('01-01-2000
```

6.4 Ipercubi

Per ogni tipologia di ipercubo è possibile creare diversi cubi, che sono aggiunti automaticamente al fondo di ogni singolo menu. Gli ipercubi definiti posso essere moficati selezionandoli da questo menu.
Per lanciare un ipercubo, occorre inserirlo in un layout di indagine.

Le tipologie di ipercubo previste sono:

- Cubi da tabelle in memoria
- Cubi da query

6.5 Funzioni

6.5.1 Nuova/Modifica

Questo comando permette di creare una nuova funzione basata su uno script o di modificarne una esistente. Il linguaggio utilizzato per gli script è il pascal script della RemObjects. Una breve guida a questo linguaggio è disponibile qui e altre informazioni sono disponibili qui.
E' disponibile un pannello di modifica dello script con basilari funzioni di debug dello stesso.
E' possibile eseguire una funzione lanciandola dall'apposito Menu Funzioni.
E' possibile anche configurare l'esecuzione automatica di una o più funzione all'apertura e alla chiusura dell'applicazione attraverso l'apposito pannello.

6.5.2 Configura esecuzione automatica funzioni

E' possibile scegliere tra le funzioni/script precedentemente definiti uno o più script da eseguire automaticamente:

- all'avvio
- alla chiusura dell'applicazione
- appena effettuata la connessione al database (e ogni volta che si cambia la connessione al database)

Questo può essere utile ad esempio per implementare funzionalità di check-controllo dei dati che devono essere eseguite all'avvio dell'applicazione.

Per creare un piano di esecuzione automatica è necessario:

- trascinare dall'elenco delle funzioni definite in alto la funzione che si desidera automatizzare col drag-n-drop nell'elenco in basso;
- selezionare l'esecuzione all'avvio e/o alla chiusura e/o alla connessione dal database selezionando gli appositi check box;
- opzionalmente impostare una priorità di esecuzione. La priorità di esecuzione serve a determinare un'ordine di esecuzione delle funzioni quando il piano ne prevede più di una. E' opzionale. Le funzioni con priorità più alta sono eseguite prima.

Per disabilitare l'esecuzione automatica di una funzione è sufficiente trascinarla fuori dall'elenco del piano di esecuzione e riportarla nell'elenco delle funzioni disponibili in alto.

6.5.3 Configura provider da funzioni

Un provider da funzione è costituito da uno script PascalScript in grado di riempire un dataset che all'interno dello script è chiamato `OutputDataset`.

Sono definiti 3 tipi di provider da funzione:

Statico:

- Viene eseguito una sola volta durante l'esecuzione dell'applicazione subito dopo la connessione al database.
- Il suo output può essere utilizzato per alimentare un report master.
- Il suo output può essere utilizzato come tabella joinabile in vari report.
- Ha un ciclo di vita ed un utilizzo simile a quello di una tabella in memoria da cdb.

Dinamico Master:

- Ha un ciclo di vita che combacia con quello di un report perchè viene eseguito ogni volta che il report viene caricato o rinfrescato.

- Il suo output può essere utilizzato per alimentare un report master.
- Nella reportistica ha un utilizzo simile a quello di una query master.

Dinamico Detail:

- Ha un ciclo di vita che combacia con quello di un report perchè viene eseguito ogni volta che un report viene caricato o rinfrescato.
- Il suo output può essere utilizzato per alimentare un report detail. In questo caso lo script può accedere alla selezione sul report master attraverso un dataset virtuale chiamato `MasterDataset`.
- Nella reportistica ha un utilizzo simile a quello di una query detail.

Nel pannello di configurazione è possibile definire un nuovo provider, definirne il tipo (statico, dinamico master, dinamico detail) e associarlo ad una funzione/rops. Dal menu contestuale è possibile creare una nuova funzione o modificare una funzione già esistente.

Per ogni provider è necessario definire la struttura dell'`OutputDataset`:
nella parte destra della finestra è visibile la struttura attuale dell'`OutputDataset`. E' possibile aggiungere una nuova colonna impostando nome, tipo e dimensione (per il tipo stringa) o eliminare una colonna già definita.
E' possibile inizializzare la struttura prelevandola da una query inserita "al volo" cliccando sul tasto "SQL" presente nella toolbar. E' possibile esportare una struttura di dataset su file e importarne una precedentemente salvata su file cliccando sui due appositi pulsanti della toolbar.

6.5.3.1 Esempio provider statico

Questo provider esegue una chiamata SOAP ad un webservice che restituisce l'elenco dei 10 film più visti al mondo e li inserisce nella tabella di output.
La chiamata SOAP è implementata in una dll esterna ad-hoc.
La tabella di output era stata precedentemente definita come costituita da 2 campi:

- `FIELD0` di tipo `string` lungo 150 caratteri
- `FIELD1` di tipo `integer`

Esempio:

```
type
  TStringArray = array of WideString;

function GetTop10:TStringArray; external 'GetTop10@"Q:\AtomosVision\SOAPTest.dll"';

var
  elenco : TStringArray;
  i : integer;
begin
  elenco := GetTop10;
  for i := 0 to 9 do
  begin
    OutputDataset.Append;
    OutputDataset.FieldByName('FIELD0').AsString := elenco[i];
    OutputDataset.FieldByName('FIELD1').AsInteger := i + 1;
    OutputDataset.Post;
  end;
end.
```

6.5.3.2 Esempio provider dinamico master

6.5.3.3 Esempio provider dinamico detail

6.5.4 PascalScript

Questo è l'elenco dei parametri in ingresso che sono comuni a tutti gli script:

Nome	Tipo	Modificabile nello script	Descrizione
CurrentUser	variabile di tipo string	no	Contiene l'utente ACN attuale
MainDBConnectionAliasName	variabile di tipo string	no	Contiene il nome dell'alias della connessione principale al database
QueryInterface	oggetto		Per l'accesso al database
FastReportInterface	oggetto		Per la stampa e l'export di report FastReport
StoredProcInterface	oggetto		Per l'accesso alle stored procedure del database

6.5.4.1 Oggetto QueryInterface

L'oggetto QueryInterface permette di scrivere una query e di eseguirla sul db corrente. La QueryInterface agisce come un repository di query, nel senso che è possibile definire più query contemporaneamente ed utilizzare l'una o l'altra attraverso il loro QueryIndex.

I metodi dell'oggetto QueryInterface sono:

```
function Prepare (aSQL : string; var QueryIndex : integer) : boolean;
```

Questa funzione crea una nuova query a partire dall'SQL passato, la prepara e restituisce l'indice della query appena creata nel repository.

Ritorna true se l'operazione ha avuto successo, false altrimenti. In caso di errore, la descrizione dell'errore può essere letta con il metodo GetLastError. L'indice ritornato serve per poter eseguire altri comandi sulla query.

```
function Open (aQueryIndex : integer) : boolean;
```

Apre la query definita all'indice passato. Ritorna true se la open ha avuto successo, false altrimenti. In questo caso la descrizione dell'errore può essere recuperata attraverso il metodo GetLastError.

```
function SetStringParameter (aQueryIndex : integer; aParameterName, aParameterValue : string) : boolean;
function SetFloatParameter (aQueryIndex : integer; aParameterName : string; aParameterValue : double) : boolean;
```

```
function SetIntegerParameter (aQueryIndex : integer; aParameterName: string; aParameterValue : integer) : boolean;
function SetDateTimeParameter (aQueryIndex : integer; aParameterName: string; aParameterValue : double) : boolean;
function SetBooleanParameter (aQueryIndex : integer; aParameterName: string; aParameterValue : boolean) : boolean;
```

Queste funzioni servono a settare gli eventuali parametri della query. Possono essere invocati subito dopo la Prepare. Restituiscono true in caso di successo e false altrimenti. In questo caso la descrizione dell'errore può essere recuperata attraverso il metodo GetLastError.

```
function SetStringParameterToNull (aQueryIndex : integer; aParameterName : string) : boolean;
function SetFloatParameterToNull (aQueryIndex : integer; aParameterName : string) : boolean;
function SetIntegerParameterToNull (aQueryIndex : integer; aParameterName : string) : boolean;
function SetDateTimeParameterToNull (aQueryIndex : integer; aParameterName : string) : boolean;
function SetBooleanParameterToNull (aQueryIndex : integer; aParameterName : string) : boolean;
```

Queste funzioni servono a settare a NULL i parametri della query. Il funzionamento è analogo a quello delle funzioni illustrate sopra.

```
function GetDataset (aQueryIndex : integer) : TDataset;
```

Dopo una open è possibile scorrere il dataset risultato della query con questo metodo.

```
function Close (aQueryIndex : integer) : boolean;
```

Chiude la query a cui è associato un certo indice. Ritorna true in caso di successo, false altrimenti. In questo caso la descrizione dell'errore può essere recuperata attraverso il metodo GetLastError.

```
procedure Delete (aQueryIndex : integer) ;
```

Elimina la query associata ad un certo indice dalla QueryInterface.

```
procedure DeleteAllQueries;
```

Elimina tutte le query presenti nella QueryInterface.

```
function ExecSQL (aQueryIndex : integer) : boolean;
```

Esegue la query associata ad un certo indice. Ritorna true in caso di successo, false altrimenti. In questo caso la descrizione dell'error può essere recuperata attraverso il metodo GetLastError.

```
function GetLastError : string;
```

Ritorna la descrizione dell'ultimo errore avvenuto.

6.5.4.1.1 Esempi

Questo script ritorna il numero di ODL presenti in una tabella:

```
var
 num, SelectIdx : integer;
 SelectSQL : string;
begin
 Error := 1;
 selectSQL := 'SELECT COUNT(*) AS NUM FROM ODL WHERE';
 selectSQL := selectSQL + '(CODICE_PARTE=:PARTE) AND (CODICE_VERSIONE=:VERSIONE)';
 if not QueryInterface.Prepare(selectSQL, SelectIdx) then
 begin
   ErrorString := 'QueryInterface.Prepare(selectSQL, SelectIdx) ' +  QueryInterface.GetLastError;
   QueryInterface.DeleteAllQueries;
   exit;
 end;

 QueryInterface.SetStringParameter(SelectIdx, 'PARTE', 'AAAXXX');
 QueryInterface.SetStringParameter(SelectIdx, 'VERSIONE', '*');
 if not QueryInterface.Open(SelectIdx) then
 begin
   ErrorString := QueryInterface.GetLastError + ' QueryInterface.Open(SelectIdx)';
   QueryInterface.DeleteAllQueries;
   exit;
 end;
 num := QueryInterface.GetDataset(SelectIdx).FieldByName('NUM').AsInteger;
 QueryInterface.Close(SelectIdx);
 ShowMessage('Ci sono ' + IntToStr(num) + ' ODL sul db intestati alla parte AAAXXX.');

 QueryInterface.DeleteAllQueries;

 Error := 0;
end.
```

6.5.4.2 Oggetto StoredProcInterface

L'oggetto StoredProcInterface consente di eseguire una STORED PROC sul db.

Il funzionamento è analogo a quello dell'oggetto QueryInterface.

I metodi dell'oggetto StoredProcInterface sono:

```
function Prepare (aStoredProcName : string; var StoredProcIndex : integer) : boolean;
```

Questa funzione prepara l'esecuzione della stored procedure che si chiama aStoredProcName e restituisce l'indice della query appena creata nel repository.

Ritorna true se l'operazione ha avuto successo, false altrimenti. In caso di errore, la descrizione dell'errore può essere letta con il metodo GetLastError. L'indice ritornato serve per poter eseguire altri comandi sulla stored proc.

```
function Execute (aStoredProcIndex : integer) : boolean;
```

Esegue la stored proc definita all'indice passato. Ritorna true se la open ha avuto successo, false altrimenti. In questo caso la descrizione dell'errore può essere recuperata attraverso il metodo GetLastError.

```
function Close (aStoredProcIndex : integer) : boolean;
```

Chiude la stored proc definita all'indice passato. Ritorna true se la open ha avuto successo, false altrimenti. In questo caso la descrizione dell'errore può essere recuperata attraverso il metodo GetLastError.

```
function SetStringParameter (aStoredProcIndex : integer; aParameterName, aParameterValue : string) : boolean;
function SetFloatParameter (aStoredProcIndex : integer; aParameterName : string; aParameterValue : double) : boolean;
function SetIntegerParameter (aStoredProcIndex : integer; aParameterName: string; aParameterValue : integer) : boolean;
function SetDateTimeParameter (aStoredProcIndex : integer; aParameterName: string; aParameterValue : double) : boolean;
function SetBooleanParameter (aStoredProcIndex : integer; aParameterName: string; aParameterValue : boolean) : boolean;
```

Queste funzioni servono a settare gli eventuali parametri della store proc. Possono essere invocati subito dopo la Prepare. Restituiscono true in caso di successo e false altrimenti. In questo caso la descrizione dell'errore può essere recuperata attraverso il metodo GetLastError.

```
function SetStringParameterToNull (aStoredProcIndex : integer; aParameterName : string) : boolean;
function SetFloatParameterToNull (aStoredProcIndex : integer; aParameterName : string) : boolean;
function SetIntegerParameterToNull (aStoredProcIndex : integer; aParameterName : string) : boolean;
```

```
function SetDateTimeParameterToNull (aStoredProcIndex : integer; aParameterName : string) : boolean;
function SetBooleanParameterToNull (aStoredProcIndex : integer; aParameterName : string) : boolean;
```

Queste funzioni servono a settare a NULL i parametri della stored proc. Il funzionamento è analogo a quello delle funzioni illustrate sopra.

```
function GetLastError : string;
```

Ritorna la descrizione dell'ultimo errore avvenuto.

6.5.4.3 Oggetto MemDBInterface

L'oggetto MemDBInterface fornisce un'interfaccia verso il CDB caricato all'avvio di AtomosVision.

I metodi dell'oggetto MemDBInterface sono:

```
Function GetTable( aTableName : string) : TDataset;
```

Questa funzione restituisce la tabella di nome aTableName presente nel CDB sottoforma di TDataset.

Questa tabella può essere letta ed eventualmente modificata utilizzando le funzioni del TDataset.

```
Procedure NotifyTableIsChanged( aTableName : string);
```

Questa funziona serve a forzare il refresh delle join dei report presenti nel layout di indagine corrente che eventualmente utilizzassero aTableName in join con i propri dati.

```
Procedure AddIndex(aTableName, aFields : string);
```

Aggiunge un indice ad una tabella del CDB.

```
Procedure RemoveIndex(aTableName : string);
```

Rimuove l'indice di una tabella del CDB.

```
Procedure RemoveAllIndexes;
```

Rimuove tutti gli indici **creati dai rops** dalle tabelle del CDB.

```
Function FindKey(aTableName : string; aValue : variant) : boolean;
```

Cerca la prima occorrenza di un certo record secondo il valore dei campi precedentementi prescelti per creare un indice.

6.5.4.3.1 Esempi

Ho una pivot master che usa la tabella 'ARTICOLI' in join con i dati estratti da una query. A questa pivot ho collegato una griglia editabile. Voglio che al commit effettuato sulla griglia editabile vengano aggiornate le informazioni relativi agli articoli impostando un valore 1 nella colonna 'MODIFICATO' e che la pivot mostri i dati aggiornati.
La struttura dello script di commit potrebbe essere la seguente:

```
var
  t : TDataset;
  articolo_da_modificare : variant;
  parte_modificata, versione_modificata : string;
  [,,,]
begin

  [... codice per la commit su db ..]

  parte_modificata := Changes.FieldByName('PT_ID_PARTE').AsString;
  versione_modificata := Changes.FieldByName('PT_ID_VERSIONE').AsString;

  [...]

  t := MemDBInterface.GetTable('ARTICOLI');
  articolo_da_modificare := CreateVarArray(2);
  AssignValueToVarArrayItem(articolo_da_modificare, 0, parte_modificata);
  AssignValueToVarArrayItem(articolo_da_modificare, 1, versione_modificata);

  if t.Locate('PT_ID_PARTE;PT_ID_VERSIONE', articolo_da_modificare, []) then
  begin
     t.Edit;
     t.FieldByName('MODIFICATO').AsInteger := 1;
    t.Post;
```

```
    MemDBInterface.NotifyTableIsChanged('ARTICOLI');
  end
  else
     ShowMessage('Errore! Articolo non trovato.');

  [...]

end.
```

Lo stesso esempio con gli indici:

```
var
  t : TDataset;
  articolo_da_modificare : variant;
  parte_modificata, versione_modificata : string;
  [,,,]
begin

  [... codice per la commit su db ..]

  parte_modificata := Changes.FieldByName('PT_ID_PARTE').AsString;
  versione_modificata := Changes.FieldByName('PT_ID_VERSIONE').AsString;

  [...]

  t := MemDBInterface.GetTable('ARTICOLI');

  MemDBInterface.AddIndex('ARTICOLI', 'PT_ID_PARTE;PT_ID_VERSIONE');

  articolo_da_modificare := CreateVarArray(2);
  AssignValueToVarArrayItem(articolo_da_modificare, 0, parte_modificata);
  AssignValueToVarArrayItem(articolo_da_modificare, 1, versione_modificata);

  if MemDBInterface.FindKey('ARTICOLI', articolo_da_modificare) then
  begin
     t.Edit;
     t.FieldByName('MODIFICATO').AsInteger := 1;
    t.Post;
    MemDBInterface.NotifyTableIsChanged('ARTICOLI');
  end
```

```
  else
     ShowMessage('Errore! Articolo non trovato.');

  [...]

  MemDBInterface.RemoveIndex('ARTICOLI');

end.
```

6.5.4.4 Oggetto FastReportInterface

L'oggetto FastReportInterface fornisce un'interfaccia verso il motore di FastReport.

I metodi dell'oggetto FastReportInterface sono i seguenti:

```
Function GetLastError : string;
```
Restituisce la descrizione dell'ultimo errore occorso

```
Procedure AddQueryParameter( aQueryName, aParamName : string; aValue : variant);
```
Valorizza il parametro di una query inserita dentro un report FastReport

```
Procedure AddVariable( aVariableName : string; aValue : variant);
```
Valorizza una variabile definita dentro un report FastReport

```
Function Print(aFileName: string): boolean;
```
Stampa o esporta un report FastReport. aFileName specifica il nome del file, senza percorsi. Il file verrà cercato dall'applicazione nelle cartelle del company chart associate all'utente ACN corrente secondo le medesime logiche utilizzate per il menu "Stampe".

Le proprietà dell'oggetto FastReportInterface sono le seguenti:

```
PrinterName: string;
```

Nome della stampante da utilizzare per la stampa del report. Se non definita, viene utilizzata la stampante predefinita di Windows corrente.

```
ExportToPDF: boolean;
```

Se true abilita l'export verso pdf del report FastReport al posto della sua stampa su carta. Il default e' false.

```
ExportFileName: string;
```

Nome del file verso cui esportare il report FastReport. Il nome deve essere completo di percorso valido. Questa opzione ha senso solo se ExportToPDF vale true.

```
ExportFileNameAddCurrentDateTime: boolean;
```

Aggiunge in coda al nome del file definito nella property ExportFileName una stringa contenente la data-ora corrente.

```
ShowPreview: boolean;
```

Mostra la preview di stampa del report invece di procedere direttamente alla stampa o all'export su file. Dalla preview di stampa l'utente puo' scegliere se stampare, se esportare o se annullare l'operazione. Il default è false.

```
Copies: integer;
```

Numero di copie da stampare. Il default è 1.

6.5.4.4.1 Esempi

Stampa di un report FastReport sulla stampante di default:

```
begin
  FastReportInterface.Print('MARIO_ETICHETTE.FRA');
```

```
end.
```

Export su pdf di report FastReport:

```
begin
  FastReportInterface.ExportToPDF := true;
  FastReportInterface.ExportFileName := 'w:\mimmo\prova.pdf';
  FastReportInterface.Print('master detail liscio.fra');
end.
```

Stampa di un report FastReport impostando il parametro di una query:

```
begin
  FastReportInterface.AddQueryParameter('NicimQuery1', 'CODICE_ORDINE', 200600041170);
  FastReportInterface.Print('REPORT_ODL.FRA');
end.
```

Mostra la preview di stampa:

```
begin
  FastReportInterface.ShowPreview := true;
  FastReportInterface.Print('MARIO_ETICHETTE.FRA');
end.
```

6.5.4.5 Funzioni varie

procedure ShowMessage(messaggio : string);

Mostra una finestra di dialogo con il messaggio passato come parametro e un pulsante OK.

Esempio:

```
ShowMessage('Ciao!');
```

function ShowYesNoMessage(messaggio : string) : integer;

Mostra una finestra di dialogo con il messaggio passato come parametro e un pulsante Si e un pulsante No.

Ritorna 1 se l'utente clicca su SI, altrimenti ritorna 0.

Esempio:

```
if ShowYesNoMessage ('Vuoi proseguire?') = 1 then
   ShowMessage('Hai scelto yes')
else
   ShowMessage('Hai scelto no');
```

function CreateVarArray(aCount: integer): variant;

Crea un variant di tipo array costituito da aCount elementi. Serve principalmente per poter utilizzare la funzione Locate del TDataset.

procedure AssignValueToVarArrayItem(var ArrayOfVariant : variant; Index : integer; Value : variant);

Assegna un valore ad un elemento di un array di variant. Serve principalmente per poter utilizzare la funzione Locate del TDataset.

Attenzione: l'indice parte da zero!

Esempio:

```
var
  ciccio : variant;
begin
```

```
  ciccio := CreateVarArray(2);
  AssignValueToVarArrayItem(ciccio, 0, 'pippo');
  AssignValueToVarArrayItem(ciccio, 1, 'pluto');
end.
```

procedure ShellExecute(Operation, FileName, Parameters, Directory: string; ShowCommand: Integer);

Chiede al sistema operativo di eseguire un comando.

Operation può valere: 'edit', 'explore', 'find', 'open', 'print'

FileName è il nome del file da eseguire/aprire o stampare

Parameters è una stringa di parametri da passare all'eventuale eseguibile specificato da FileName

Directory specifica le directory di default

ShowCommand specifica come deve essere visualizzata la finestra dell'applicazione eseguita (0=Nascosta, 1=Normale, 3=Massimizzata, 6=Minimizzata)

Esempio:

```
ShellExecute('open', 'w:\mimmo\prova.bat', '', '', 1);
```

function DebugMode : integer;

Ritorna 1 se la modalità di debug (Eurekalog) è attiva (menu ?->Modalità di debug), 0 altrimenti.

Può essere utile per pilotare l'accensione e lo spegnimento di qualche funzionalità di debug o di log realizzata all'interno del rops.

function Min(A: Double; B: Double): Double;

Ritorna il minore dei due valori.

function Max(A: Double; B: Double): Double;

Ritorna il massimo dei due valori.

6.5.4.5.1 Gestione dei cookies

In tutti i rops è possibile definire dei cookie strutturati in questo modo:

Chiave : string

Valore : variant

Esistono diverse categorie di cookies:

- Cookies di sessione
- Cookies di report
- Cookies di layout
- Cookies di applicazione

Non tutte le categorie di cookies sono disponibili contemporaneamente. La loro presenza dipende dal contesto in cui il rops è utilizzato. Ad esempio i cookies di report sono disponibili solo negli script delle griglie editabili.

6.5.4.5.1.1 Cookies di sessione

I cookie di sessione hanno un ciclo di vita che corrisponde a quello dei dati caricati nella griglia editabile. Quindi a partire da un refresh dei dati o da una commit dentro la griglia editabile fino al successivo refresh o commit delle modifiche. Lo scope è quello della griglia editabile.

Le funzioni richiamabili da dentro i rops sono le seguenti:

function SessionCookieExists(aCookie : string) : boolean;

Ritorna true se un certo cookie esiste.

function GetSessionCookie(aCookie : string) : variant;

Ritorna il valore di un cookie.

procedure SetSessionCookie(aCookie : string; aValue : variant);

Imposta il valore di un cookie. Se il cookie esiste già, il valore precedente viene sovrascritto. Se il cookie non esiste, viene creato.

procedure ClearSessionCookies;

Elimina tutti i cookie di sessione definiti.

procedure ClearSessionCookiesLike(aPattern: string);

Elimina tutti i cookie di sessione la cui chiave combacia col pattern definito.

Esempio:

`ClearSessionCookiesLike('attributo_domanda_1')` elimina il cookie "attributo_domanda_1"

`ClearSessionCookiesLike('attributo_domanda_%')` elimina tutti i cookie il cui nome inizia con "attributo_domanda_"

`ClearSessionCookiesLike('%_1')` elimina tutti i cookie il cui nome termina con "_1"

6.5.4.5.1.2 Cookies di report

I cookie di report hanno un ciclo di vita che corrisponde a quello della griglia editabile. Quindi a partire dal caricamento del layout di indagine o dalla sua creazione fino alla chiusura del layout o alla chiusura dell'applicazione. Lo scope è quello del singolo report.

Le funzioni richiamabili da dentro i rops sono le seguenti:

function ReportCookieExists(aCookie : string) : boolean;

Ritorna true se un certo cookie esiste.

function GetReportCookie(aCookie : string) : variant;

Ritorna il valore di un cookie.

procedure SetReportCookie(aCookie : string; aValue : variant);

Imposta il valore di un cookie. Se il cookie esiste già, il valore precedente viene sovrascritto. Se il cookie non esiste, viene creato.

procedure ClearReportCookies;

Elimina tutti i cookie di report definiti.

procedure ClearReportCookiesLike(aPattern: string);

Elimina tutti i cookie di report la cui chiave combacia col pattern definito.

6.5.4.5.1.3 Cookies di layout

I cookie di layout hanno un ciclo di vita che corrisponde a quello del layout di indagine. Quindi a partire dal caricamento del layout di indagine o dalla sua creazione fino alla chiusura del layout o alla chiusura dell'applicazione. Lo scope è quello di tutti i report all'interno di un layout di indagine.

Le funzioni richiamabili da dentro i rops sono le seguenti:

function LayoutCookieExists(aCookie : string) : boolean;

Ritorna true se un certo cookie esiste.

function GetLayoutCookie(aCookie : string) : variant;

Ritorna il valore di un cookie.

procedure SetLayoutCookie(aCookie : string; aValue : variant);

Imposta il valore di un cookie. Se il cookie esiste già, il valore precedente viene sovrascritto. Se il cookie non esiste, viene creato.

procedure ClearLayoutCookies;

Elimina tutti i cookie di layout definiti.

procedure ClearLayoutCookiesLike(aPattern: string);

Elimina tutti i cookie di layout la cui chiave combacia col pattern definito.

6.5.4.5.1.4 Cookies di applicazione

I cookie di applicazione hanno un ciclo di vita che corrisponde a quello dell'applicazione. Quindi a partire dall'apertura dell'applicazione fino alla chiusura dell'applicazione. Lo scope è quello di tutti i rops eseguiti all'interno dell'applicazione.

Le funzioni richiamabili da dentro i rops sono le seguenti:

function ApplicationCookieExists(aCookie : string) : boolean;

Ritorna true se un certo cookie esiste.

function GetApplicationCookie(aCookie : string) : variant;

Ritorna il valore di un cookie.

procedure SetApplicationCookie(aCookie : string; aValue : variant);

Imposta il valore di un cookie. Se il cookie esiste già, il valore precedente viene sovrascritto. Se il cookie non esiste, viene creato.

procedure ClearApplicationCookies;

Elimina tutti i cookie di applicazione definiti.

procedure ClearApplicationCookiesLike(aPattern: string);

Elimina tutti i cookie di applicazione la cui chiave combacia col pattern definito.

6.5.4.6 Manipolazione di date

All'interno dei rops è definito un tipo TDateTime che rappresenta una data-ora.

Le funzioni per manipolare le date disponibili sono:

```
function EncodeDate(Year, Month, Day: Word): TDateTime;
function EncodeTime(Hour, Min, Sec, MSec: Word): TDateTime;
function TryEncodeDate(Year, Month, Day: Word; var Date: TDateTime): Boolean;
function TryEncodeTime(Hour, Min, Sec, MSec: Word; var Time: TDateTime): Boolean;
procedure DecodeDate(const DateTime: TDateTime; var Year, Month, Day: Word);
procedure DecodeTime(const DateTime: TDateTime; var Hour, Min, Sec, MSec: Word);
function DayOfWeek(const DateTime: TDateTime): Word;
function Date: TDateTime;
function Time: TDateTime;
function Now: TDateTime;
function DateTimeToUnix(D: TDateTime): Int64;
function UnixToDateTime(U: Int64): TDateTime;
function DateToStr(D: TDateTime): String;
```

```
function StrToDate(const s: String): TDateTime;
function FormatDateTime(const fmt: String; D: TDateTime): String;
```

6.5.4.7 Allarmi e notifiche

All'interno dello script è possibile mostrare degli allarmi/notifiche associate ad una icona nella barra di stato dell'applicazione e ad un messaggio testuale.

procedure ShowInfo (aMsg : string; aRemoveAfterRead : boolean);

Mostra una notifica con una icona "Info". aRemoveAfterRead va impostato a true se si desidera che il pulsante della notifica venga eliminato una volta letto il messaggio di testo.

procedure ShowWarning (aMsg : string; aRemoveAfterRead : boolean);

Mostra una notifica con una icona "Warning". aRemoveAfterRead va impostato a true se si desidera che il pulsante della notifica venga eliminato una volta letto il messaggio di testo.

procedure ShowError (aMsg : string; aRemoveAfterRead : boolean);

Mostra una notifica con una icona "Errore". aRemoveAfterRead va impostato a true se si desidera che il pulsante della notifica venga eliminato una volta letto il messaggio di testo.

procedure ShowNotification (aMsg : string; aIconFile : string; aRemoveAfterRead : boolean);

Mostra una notifica con una icona letta da file. aRemoveAfterRead va impostato a true se si desidera che il pulsante della notifica venga eliminato una volta letto il messaggio di testo.
aIconFile deve indicare un file bmp contenente una bitmap di dimensione 16x16 pixel. Il file può essere specificato indicando un nome file completo di path oppure con il solo nome del file (es. PIPPO.BMP), in questo caso il file verrà cercato nelle cartelle di ACN a partire da quella dell'utente/nodo.

6.5.4.8 Utilizzo di dll esterne

All'interno di un file rops è possibile richiamare una o più funzioni disponibili in una dll esterna.

Esempio:

```
procedure GetRevision(var aRevision : integer); external 'GetRevision@"w:\mimmo\DllNicimScheduleInterface.dll" stdcall';

var
  revisione : integer;
begin
  GetRevision(revisione);
  ShowMessage('Revisione da dll:' + IntToStr(revisione));
end.
```

6.6 Report di analisi

6.6.1 Nuovo/Modifica

Questo comando permette di creare o modificare un report di analisi. Un nuovo layout o un layout già presente modificato possono essere successivamente salvati tramite l'icona del disco posta sulla barra degli strumenti della finestra che contiene il layout stesso così come è possibile sovrascrivere un report già esistente.

I report di analisi nuovi sono richiamabili anche dal menu dei report, perché vengono automaticamente inseriti nel menu.

Successivamente possono essere inseriti nel sottomenu di competenza, tramite il comando "Organizzazione del Menu".

6.7 Report per stampe

6.7.1 Nuovo/Modifica

Questo comando permette di creare un nuovo report di stampa o di modificarne uno esistente.

I report di stampa nuovi sono richiamabili anche dal menu delle stampe, perché vengono automaticamente inseriti nel menu.

Successivamente possono essere inseriti nel sottomenu di competenza, tramite il comando "Organizzazione del Menu".

Parte VII

7 Menu ?

Questo menu contiene i comandi relativi alla guida e al debug.

7.1 Guida in linea

Apre questa guida in linea.

7.2 Informazioni su

Mostra le informazioni sulla versione dell'applicativo attualmente installata.

Lanciando questo comando tenendo premuto il tasto CTRL è possibile visionare una finestra di debug con l'elenco delle licenze Nicim attualmente associate all'utente corrente.

7.3 Modalità di debug

Pone l’applicazione in una speciale modalità di funzionamento chiamata modalità di debug.

In questa modalità l’applicazione attiva una funzionalità di log avanzata che fornisce informazioni dettagliate fondamentali per identificare eventuali problemi durante l’esecuzione del programma.

In caso di errore viene aperta una finestra di errore da cui è possibile copiare ed incollare in una mail o in un documento di testo un log dettagliato che mostra lo stato interno dell’applicazione (istruzioni eseguite, stato della memoria, thread attivi, ecc..).
Queste informazioni sono preziose per identificare possibili bachi e dovrebbero essere inoltrate all’assistenza Atomos.

Parte VIII

8 Report di analisi (concetti generali)

I report di analisi (o layout di indagine) sono lo strumento di rappresentazione ed analisi del dato.

8.1 Navigazione tra i dati

I report di analisi (o layout di indagine) rappresentano uno strumento completo di navigazione dei dati. Un layout d'indagine contiene più report tra loro collegati, secondo una relazione di tipo padre-n figli.
Ogni volta che si salva un layout, vengono anche salvate le eventuali modifiche apportate a ogni singolo scenario dei reports appartenenti al layout.

Report Padre - Report Figlio

Il layout viene costruito a partire da uno o più report padre. Le informazioni contenute nel report sono sensibili al click del pulsante destro del mouse, che nel menù associato presenta la voce Collega a... A seconda del report di partenza, l'elenco dei report collegabili è diverso. Selezionandone uno, questo viene aperto e può essere posizionato a piacere sul layout (affiancato, sovrapposto con linguetta,…). Ad un report padre possono essere collegati quanti report figli si vuole, a partire dallo stesso tipo di dato o da dati diversi. Ogni report figlio a sua volta può diventare report padre, in una catena senza limiti (se non la leggibilità…).

Allineamento Dati

Di default, il report figlio è automaticamente aggiornato coi nuovi dati, ogni qual volta si seleziona un diverso record sul report padre. Nel caso particolare in cui non si voglia che il programma automaticamente aggiorni un figlio, basterà scollegarlo utilizzando il bottone Connetti/Disconnetti presente sulla sua barra del titolo.

Barra dei Tempi

Se sono presenti più report contenenti una barra temporale (ad esempio più Gantt), lo spostamento nel tempo su un report, implica

l’automatico spostarsi degli altri report. L’effetto è particolarmente apprezzabile se tali report vengono posizionati sul layout affiancati verticalmente.

8.2 Toolbar

I pulsanti visibili nella toolbar possono variare in funzione della configurazione dell’applicazione e del profilo dell’utente corrente. I pulsanti illustrati qui sotto sono quelli di default.

Opzioni

Attraverso il pulsante è possibile accedere al pannello delle opzioni del layout di indagine.

Il pannello è diviso in due sezioni:

- Generale. Si può modificare la descrizione del layout che compare nella barra del titolo dell’applicazione quando il layout viene visualizzato. E’ possibile inserire anche una descrizione estesa del layout allo scopo di dettagliare il funzionamento del report a beneficio degli utenti che lo utilizzeranno. La descrizione estesa viene importata da un file html esterno.
- Barra dei tempi. Qui si configura la barra dei tempi (e quindi il bucket temporale) dei report appartenenti al layout. La barra ha una scala principale e una secondaria su cui possono essere evidenziati in rosso o altro colore i sabati e le domeniche. Alcuni applicativi che caricano il calendario aziendale possono evidenziare anche le festività lì riportate.

Carica layout

Il pulsante serve ad aprire un layout di indagine precedentemente salvato su disco.

I layout che appartengono al livello del company chart dell’utente corrente sono identificati dall’icona . I layout contrassegnati dall’icona appartengono ai livelli più alti del company chart.

Salva layout

Se l’applicazione è in Modalità Configuratore il pulsante salva un layout su disco. Il layout viene sempre salvato al livello del company chart a cui appartiene l’utente corrente o sul livello su cui si è spostato preventivamente usando il comando “Impersonifica nodo-utente”.

Salva opzioni personali

Se l’applicazione non è in Modalità Configuratore il pulsante salva localmente le impostazioni personali che l’utente ha modificato sul layout corrente.

Aggiungi report

Questo pulsante presenta un menu a tendina con l’elenco di tutti i report radice che possono essere aggiunti al layout di indagine. I report inseriti potranno essere usati come capostipiti di nuove catene di report padre-figlio.

Utilità

Il pulsante presenta un menu a tendina con un elenco di utilità:

- Nascondi le toolbar. Minimizza le toolbar di tutti i report presenti nel layout corrente. Durante il salvataggio di un layout viene memorizzato anche lo stato delle toolbar per poter essere ripristinato al successivo caricamento
- Struttura del layout corrente. In questo pannello viene visualizzata la gerarchia dei legami padre-figlio tra i report del layout corrente. La visualizzazione è finalizzata ad operazioni di debug e quindi di interesse solo per un utente configuratore.

Log delle decorazioni e delle colonne calcolate

Quando si inserisce una decorazione è necessario esprimere una condizione di decorazione attraverso una formula. Analogamente quando si crea una colonna calcolata in un report è necessario inserire una formula.
Può capitare che l'utente inserisca una formula errata o che inserisca una formula che funziona quasi sempre ma che alle volte, per qualche dato particolare, fallisce.

Le applicazioni sono costruite in modo tale da non bloccarsi quando una formula è errata (le formule di decorazione vengono eseguite molte volte durante il disegno a schermo e non è opportuno che vengano sollevati messaggi di errore durante questa delicata fase).
Gli errori vengono registrati in uno speciale log che può essere visualizzato cliccando sul pulsante . Se l'icona del pulsante si presenta ingrigita vuol dire che non ci sono errori presenti nel log.
Poiché le formule vengono risolte solo quando serve, l'icona potrebbe "accendersi" non solo al caricamento del report ma anche durante il suo utilizzo (quando cioè un dato critico manda in crisi una formula).
Questa funzionalità è particolarmente utile durante la fase di creazione e modifica dei report di analisi.

8.3 Barra del titolo

Di default ogni report all'interno di un layout ha una barra del titolo che riporta il nome del report e una serie di pulsanti.

Pulsante : mostra la finestra di modifica delle impostazioni personali per il report corrente. Questa finestra è accessibile solo quando l'applicazione non si trova in Modalità Configuratore.
La finestra varia da report a report ma in generale presenta l'elenco delle colonne del report (di cui è impostabile l'attributo di visibilità) e la maschera di modifica del font utilizzato nel report.

Pulsante : disconnette (e riconnette) il report dal report padre (ove presente) in modo che la navigazione tra i dati del report padre non causi il refresh dei dati nel report corrente.

Pulsante : apre un menu popup che contiene alcune funzioni per il posizionamento del report nel layout ed il comando "Rename" che serve a modificare la descrizione del report visibile nella sua barra del titolo

Pulsante : abilita/disabilita l'autohide che è la capacità del report di nascondersi automaticamente quando non è utilizzato

Pulsante X : chiude il report. Utilizzabile solo quando si è in fase di creazione o modifica del layout di indagine.

8.4 Configurazione generale dei report

8.4.1 Elementi comuni

Ogni applicativo offre la possibilità di creare diversi tipi di report su un proprio set di dati.

Esistono diverse tipologie di report: griglie, pivot, alberi, gantt, cubi, ecc.. Essi condividono alcuni elementi di configurazione.

La configurazione di un report è organizzata in scenari. Uno scenario rappresenta un pacchetto di opzioni grafiche e di visualizzazione che viene applicato ai dati visualizzati. L'utente può passare da uno scenario ad un altro attraverso un menu a tendina che solitamente si trova a sinistra nella toolbar del report.

Gli scenari sono costituiti da set di configurazioni divise per tipologia.
Le più comuni sono:

- le **viste** che racchiudono le informazioni relative alle proprietà delle colonne del report (etichetta, larghezza, visibilità, formato, ordine)
- i **filtri** che servono a filtrare i dati visualizzati nel report
- le **decorazioni** che servono per modificare l'apparenza grafica del dato visualizzato secondo condizioni impostate a piacere

All'interno del sistema di reportistica gli scenari e gli elementi che essi contengono sono condivisi tra le istanze dello stesso report. Ad esempio tutte le griglie del piano di lavoro accederanno allo stesso set di scenari (e di viste, decorazioni...) così come tutte le pivot alimentate dalla query chiamata "Query XYZ" condivideranno il medesimo set di scenari, e via così.

Altre configurazioni, sebbene non siano organizzate in scenari, sono comunque condivise tra report dello stesso tipo. Alcuni esempi di queste configurazioni sono:

- le join
- le colonne calcolate
- le barre calcolate dei gantt
- le formule su raggruppato delle pivot
- ecc..

8.4.2 Modifica della configurazione

In Modalità Configuratore il pannello di modifica della configurazione di un report è solitamente accessibile da un pulsante sulla toolbar con questo aspetto:

Il pannello è organizzato in 2 aree:

- a sinistra è riportato l'elenco dei tipi di set di configurazioni editabili (viste, scenari, colonne custom, ecc..)
- navigando nell'elenco di sinistra, sulla destra vengono presentati dei pannelli di configurazione specifici per il tipo di configurazione.

Gli elementi visualizzati con sfondo di colore rosa sono elementi di configurazione ereditati da un livello del company chart più alto del livello di appartenenza dell'utente e come tali non possono essere modificati ma solo riutilizzati per la visualizzazione o, ad esempio, per la composizione di nuovi scenari.

Quando l'applicazione non è in Modalità Configuratore il pannello di modifica della configurazione di un report è solitamente accessibile dal pulsante presente sulla barra del titolo del report.
Questo pulsante permette l'accesso ad un pannello di configurazione semplificato rispetto al precedente.
Tutte le modifiche effettuate vengono associate allo scenario attuale e vengono sovrascritte alla configurazione ricavata dallo scenario.

Il pulsante Ripristina default ripristina la configurazione così come ricavata dallo scenario.

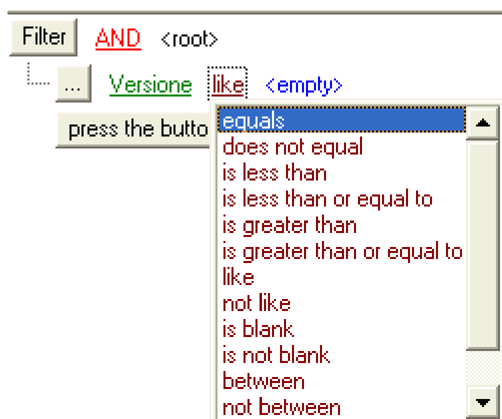
8.4.2.1 Configurazione dei filtri

È possibile creare più filtri da associare a scenari diversi o da richiamare durante la visione della tabella.

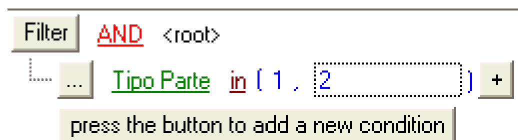
1. Cliccando sulla voce AND si può scegliere l'operatore con il quale collegare tra loro le condizioni (AND, OR, NOT AND, NOT OR).
2. Per aggiungere una nuova condizione cliccare su press the button to add a new condition.
3. A questo punto si sceglie il campo su cui operare la condizione, cliccare sulla voce in verde (primo campo della tabella).



4. Scegliere la condizione tra quelle a disposizione (cliccare su like).



5. Nel campo in azzurro definire il valore per il filtro, nell'esempio seguente prendiamo tutti i tipo parte 1 e 2.

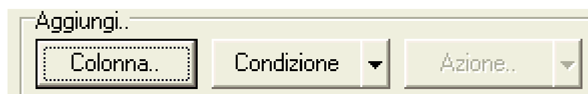


In alcuni report un filtro si può realizzare anche direttamente dal report: cliccare sulla freccetta sulla intestazione della colonna e scegliere il valore voluto o selezionare la voce *Custom*.

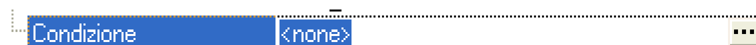
Per i campi oggetto di filtro tale freccetta diventa azzurra Tipo Parte .

8.4.2.2 Configurazione delle decorazioni di celle

Le *Decorazioni* all'interno di uno scenario definiscono l'arricchimento grafico nella visualizzazione dei dati.
È possibile creare più modalità di colorazione da associare a scenari diversi.

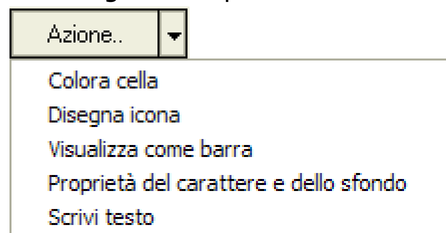


1. Tasto Colonna per definire il campo della tabella che si vuole decorare.
2. Tasto Condizione per definire la condizione



cliccare una volta sul campo <none> e poi sul tastino che compare a destra. È ora possibile scegliere il campo che deve soddisfare la condizione ed eventuali formule messe a disposizione del sistema.

3. Scegliere il tipo di colorazione



Colora cella: colora la cella in funzione della condizione impostata (il testo originale non è più visibile).

Disegna icona: inserisce una bitmap se sono verificate le condizioni impostate.
Visualizza come barra: disegna una barretta della lunghezza del valore della cella; si deve indicare anche il minimo e massimo valore della scala.
Proprietà del carattere e dello sfondo: per definire il colore dello sfondo della cella e la proprietà del carattere (font, dimensione, colore).
Scrivi testo: permette di inserire una scritta personalizzata al verificarsi di una condizione.

8.4.2.3 Configurazione delle join

Si utilizza per collegare tabelle esterne (caricate in memoria attraverso il CDB), da cui attingere campi da visualizzare o da usare nelle condizioni.
Si indica la tabella di interesse (tra quelle caricate in memoria), i campi che le legano e i campi che si vogliono utilizzare.

Solitamente i campi ricavati da una join assumono un nome del tipo NomeTabellaInJoin_NomeCampo per essere più facilmente identificati.

8.4.2.4 Configurazione delle colonne custom

Il sistema permette di definire dei nuovi campi da visualizzare nel report o da utilizzare nelle condizioni.
Si dovrà indicare il nome del campo, l'etichetta con cui visualizzarlo, il tipo (numero, data, stringa) e la formula con cui calcolarlo.
Per la sua composizione è possibile combinare tra loro i vari campi della tabella, utilizzando le formule che il sistema mette a disposizione.
I campi utilizzabili sono tutti quelli del report, compresi quelli derivati da join.

Le colonne custom appena definite sono disponibili alla successiva riapertura del report.

8.4.2.5 Configurazione dei font

E' possibile variare le proprietà dei caratteri utilizzati in un report impostando un valore per il font, la dimensione e gli eventuali attributi di formattazione (grassetto, corsivo, ecc..).
Nella finestra di modifica in basso è visibile un box con l'anteprima del risultato.

8.5 Report griglia

I report griglia forniscono una rappresentazione tabellare dei dati.

Di default i dati vengono rappresentati come in una matrice di righe e colonne.

Ordinamento

La griglia permette all'utente di ordinare i dati. L'utente può modificare l'ordinamento dei valori di una colonna cliccando sull'intestazione della colonna. Il primo click ordinerà i valori in ordine crescente, il secondo in ordine decrescente. Per eliminare un criterio di ordinamento l'utente deve cliccare sull'intestazione della colonna tenendo premuto il tasto CTRL.
E' possibile ordinare più colonne contemporaneamente, per far questo bisogna tener premuto il tasto SHIFT cliccando sulle intestazioni delle colonne.
Una piccola freccia all'interno dell'intestazione di una colonna indica qual e' l'attuale ordinamento. Se la freccia punta verso l'alto l'ordinamento è crescente, se verso il basso l'ordinamento è decrescente.

Drag a column header here to group by that column

Trademark	Model
BMW	530i
Cadillac	Seville
Cadillac	DeVille
Dodge	Ram 1500
Ford	Ranger FX-4
GMC	Siera Quadrasteer
Jaguar	S-Type 3.0
Lexus	LS430
Lexus	GS 430
Mercedes-Benz	SL500 Roadster
Mercedes-Benz	CLK55 AMG Cabriolet
Mercedes-Benz	C230 Kompressor Sport Coupe
Nissan	Crew Cab SE
Rolls-Royce	Corniche
Toyota	Tacoma S-Runner

Filtri

La griglia permette di filtrare i dati visualizzati attraverso la definizione di filtri. Quando un filtro è applicato, nel report sono visibili solo quei dati che soddisfano la condizione di filtro.

Un tipo "rapido" di filtro può essere creato in modo molto veloce. Di default ogni colonna ha un pulsante con una freccia a destra della sua intestazione. Cliccando su questo pulsante viene visualizzato un elenco con i valori unici contenuti nella colonna corrente.

Drag a column header here to group by that column

Trademark	Model
(All)	530i
(Custom...)	Seville
BMW	DeVille
Cadillac	Ram 1500
Dodge	Ranger FX-4
Ford	Siera Quadrasteer
GMC	S-Type 3.0
Jaguar	LS430
Lexus	GS 430
Mercedes-Benz	SL500 Roadster
Nissan	CLK55 AMG Cabriolet
Rolls-Royce	C230 Kompressor Sport Coupe
Toyota	Crew Cab SE
Mercedes-Benz	
Mercedes-Benz	
Mercedes-Benz	
Nissan	
Rolls-Royce	Corniche
Toyota	Tacoma S-Runner

Quando l'utente seleziona un valore da questa lista, un nuovo filtro del tipo "Colonna = Valore" è creato. Selezionando la voce "All" il filtro viene annullato.

E' anche possibile definire dei filtri più complessi con espressioni articolare e diversi operatori (=, <>, <, <=, >, >=, LIKE and NOT LIKE).

Raggruppato

Quando i dati vengono raggruppati, i record con il medesimo valore nella colonna raggruppata formano un gruppo e vengono visualizzati nel report come fossero nodi di un albero.

Project | Type | Status

Issue Info				Creator Info	
Name	Priority	Fixed Dat	Created D	Creator Name	Creator Pho
+ Project : Easy life insurance: Agent Edition					
− Project : Easy car insurance: Agency Edition (enterprise version)					
− Type : Bug					
− Status : New					
Commisions calculator: Chargebacks	Low	8/31/2002	4/3/2002	Jimmy Lewis	(171)55-12-12
Commissions table: Blinking	Hi	8/16/2002	6/26/2002	Dave Murrel	452-07-65
Email: Fax	Low	8/1/2002	5/9/2002	Dave Murrel	452-07-65
Emails: High Contrast black mode	Low	8/25/2002	7/12/2002	Dave Murrel	452-07-65
Help: Duplicate Campaign window	Low	8/8/2002	5/10/2002	Mark Hamilton	912-44-40
Help: Commissions Profile window	Low	8/11/2002	5/14/2002	Jimmy Lewis	(171)55-12-12
+ Status : Rejected					
+ Type : Request					
+ Project : Insurance adjuster: Agency Edition					

Per raggruppare i dati l'utente deve trascinare col mouse l'intestazione della colonna (che si desidera far diventare la chiave del raggruppato) nel "Group by box" (quell'area griglia che compare in cima alla griglia). Si possono definire più raggruppamenti in cascata (ad esempio raggruppo la tabella delle commesse per cliente e poi per data di consegna).
Il raggruppato può essere annullato trascinando l'instestazione della colonna dal "Group by box" indietro dove compaiono le instesazioni

delle colonne.

Quando un raggruppato è attivo, i dati vengono organizzati su livelli.

Ad esempio raggruppando per 2 campi, 3 livelli vengono visualizzati:

Totali

Totali e subtotali possono essere calcolati su di un gruppo di record o su tutti i record della griglia.

Ci sono due tipi di totali: i totali del footer e i totali del raggruppato.

CustomerID

PurchaseDate	PaymentType	PaymentAmount	Quantity
+ CustomerID : Mickeys World of Fun (Records: 6)			
− CustomerID : Michelson Systems (Records: 5)			
2/20/2002	Master	$76,800.00	3
1/22/2002	Cash	$118,350.00	3
2/14/2002	Visa	$1,852,425.00	5
2/1/2002	Am. Express	$164,968.00	4
1/31/2002	Cash	$64,000.00	5
MAX=2/20/2002		SUM = $2,276,543	
Total Records = 11			

I totali dell'area footer sono calcolati su tutti i record della griglia. Quelli del raggruppato solo sui record che appartengono al raggruppato.

E' possibile definire diversi tipi di totali calcolati in modo, ad esempio, che venga visualizzato il minimo o il massimo o la somma o la media dei valori presenti in una certa colonna.

Ricerca incrementale

La ricerca incrementale permette all'utente di localizzare le righe della griglia sulla base dei valori di una certa colonna.
L'utente, scrivendo in una colonna, viene trasportato sul primo record della colonna che inizia con le lettere digitale.

I caratteri 'mi' sono stati digitali. Come risultato il primo 'Mickey Alcorn' viene localizzato.

Il carattere 'l' è premuto dopo 'mi' e il record corrente diventa 'Mildred Johanson'.

Drag a column header here to group by that column

Customer	PaymentType	PaymentAmount
Karen Holmes	Cash	$83,800.00
Mickey Alcorn	Visa	$251,400.00
Mickey Alcorn	Master	$251,400.00
Jimmie Jones	Am. Express	$335,200.00
Jimmie Jones	Master	$83,800.00
Bobbie Valentine	Cash	$167,600.00
Mildred Johansson	Cash	$251,400.00
Bobbie Valentine	Am. Express	$419,000.00

Drag a column header here to group by that column

Customer	PaymentType	PaymentAmount
Karen Holmes	Cash	$83,800.00
Mickey Alcorn	Visa	$251,400.00
Mickey Alcorn	Master	$251,400.00
Jimmie Jones	Am. Express	$335,200.00
Jimmie Jones	Master	$83,800.00
Bobbie Valentine	Cash	$167,600.00
Mildred Johansson	Cash	$251,400.00
Bobbie Valentine	Am. Express	$419,000.00

Cliccando su Ctrl+UpArrow o Ctrl+DownArrow l'utente può spostarsi sui record che soddisfano i valori digitati verso l'alto o verso il basso.

8.5.1 Operatività griglia

Menù Tasto Destro da Intestazione Campi

Con il cursore sulla riga di intestazione delle colonne, con il tasto destro del mouse si richiama il menù per modificare il layout della tabella.
Sort: consente l'ordinamento della tabella in funzione dei valori di una colonna. Questa funzione è anche attivabile esternamente al menù, cliccando con il tasto sinistro del mouse sulla intestazione del campo di interesse. Un clic per l'ordinamento crescente, due per il decrescente. Tenendo poi premuto il tasto Shift, si può ordinare una tabella in funzione di più colonne. Per i campi sortati compare un freccia grigia sull'intestazione, in su o in giù a seconda se è ascendente o discendente.
E' possibile eliminare l'ordinamento impostato con il tasto ***Clear Sorting***.

Group By: sia con questa voce di menù sia trascinando il campo voluto nell'area grigia in cima alla tabella, è possibile creare dei raggruppamenti logici. Se si trascina nuovamente il campo raggruppato sulla griglia di tabella, viene eliminato il raggruppamento.

Group By Box: tale tasto evidenzia uno spazio grigio in cima alla tabella in cui trascinare le colonne oggetto di raggruppamento.

Footer: inserisce una riga in fondo alla tabella, nella quale è poi possibile riportare, per ogni colonna, il risultato di alcune operazioni. Il tipo di informazione da visualizzare si sceglie posizionandosi sul footer all'altezza della colonna desiderata.

Group Footers: stessa cosa del Footer ma per i campi raggruppati.

Remove this column: per non visualizzare le colonne indesiderate.

Field chooser: apre un form contenente le colonne eliminate, che permette il loro ripristino trascinandole sulla tabelle nel punto voluto.

Best fit: ottimizza la larghezza della colonna in funzione delle informazioni che contiene. È possibile farlo per tutte le colonne contemporaneamente (***All columns***).

Menù Tasto Destro da Riga Tabella

Con il cursore su una qualsiasi riga della tabella, il tasto destro del mouse richiama il menù in oggetto.

Copia negli appunti: fa una copia del testo contenuto nella cella selezionata. Attivabile anche con la combinazione di tasti Ctrl C.

Visualizza/nascondi footer del gruppo: permette di visualizzare o meno le eventuali operazioni del group footer anche nel caso di gruppo non esploso.

Seleziona i record del gruppo: posizionandosi su una qualsiasi riga di un gruppo, permette di selezionare tutti i record appartenenti a quel gruppo.

Carica / Salva Configurazione

E' data la possibilità di salvare una configurazione impostata (tasto) e successivamente richiamarla con il tasto Carica Configurazione ( ).

Espandi/Collassa – Ridimensionamento colonne

In caso di raggruppamenti attivi, è possibile espandere tutti i rami relativi ai raggruppamenti ( ) oppure collassarli tutti ( ).

E’ inoltre disponibile il tasto di ottimizzazione della dimensione delle colonne della tabella in oggetto ( ).

Utilità di Stampa / Esportazione tabella

E’ possibile inviare ad una coda di stampa la vista selezionata, come pure salvare la tabella su file ( ).
I formati disponibili sono: excel, html, xml, file di testo.

Opzioni Configurazione

Attivabile tramite il tasto *Configura* della barra degli strumenti.

Viste

Le *Viste* all'interno di uno scenario definiscono le configurazioni delle colonne (caption, formato e visibilità).
È possibile creare più viste da associare a scenari diversi. La voce *Clona* crea una esatta copia della vista selezionata.
Colonna: per definire il nome del campo da visualizzare in tabella
Visibile: per decidere se visualizzare o meno il campo
Formato: per definire il formato del campo (per valori numerici)
Campo: non editabile, riporta il vero nome del campo nella tabella.
Lo sfondo bianco delle cellette significa che è stata modificata la configurazione di default.

Layout Scenario

È possibile associare ad ogni scenario un particolare menù
Per modificarlo basta richiamare lo scenario che lo contiene, fare le modifiche del caso e poi salvare il report.
Il layout definisce l’organizzazione del dato in tabella (group by, sort, larghezza colonne, footer …).

Scenari Griglia

È possibile definire diversi scenari di visualizzazione, richiamabili dal menù a tendina della toolbar.
Ad ogni scenario è associata una particolare *Vista*, *Filtro*, *Decorazione* e *Layout*.
Da una stessa tabella è quindi possibile ricavare reports diversi, in funzione degli scenari configurati. Per esempio dai Movimenti Simulati posso ricavare un report focalizzato sui solleciti di materia prima e uno sugli ordini di produzione.

8.5.2 Subtotali

Su di una griglia con almeno un raggruppamento è possibile permette di definire un subtotale per gruppo da visualizzare nell'intestazione del gruppo oltre che nell'area footer del gruppo.

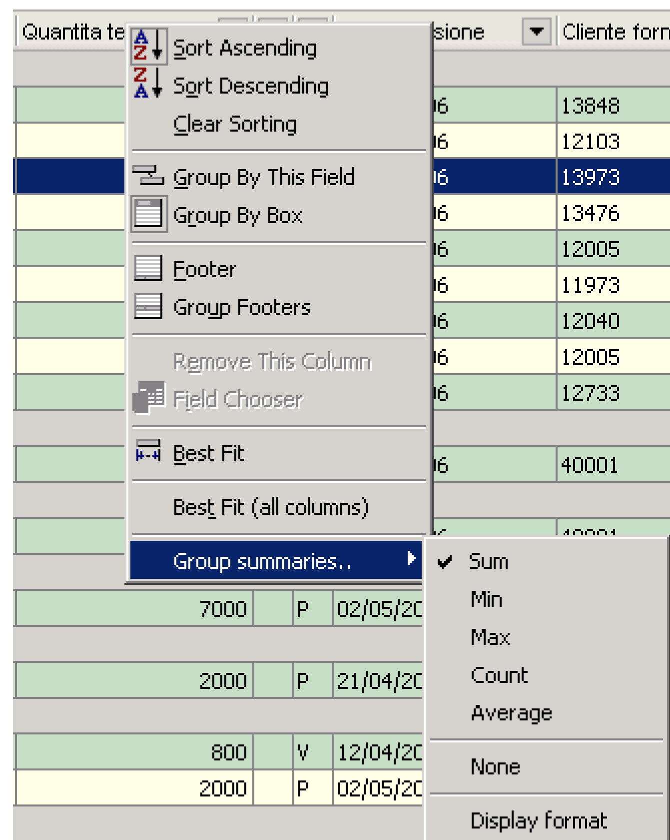
Griglia tabella in memoria_1

Scenario2 | Filtro

Codice parte

Codice ordine	Codice f	Cc	Data disp	Codice commessa	Codice m		Quantita teorica
Codice parte : 0101118320501 (Quantita teorica SUM=17700)							
IF-794.001	0		20/04/2006	IF-794.001	01	-	200
IF-733.001	0		18/04/2006	IF-733.001	01	-	500
IF-672.002	0		11/04/2006	IF-672.002	01	-	1000
IF-781.002	0		20/04/2006	IF-781.002	01	-	1000
IF-730.001	0		18/04/2006	IF-730.001	01	-	2000
IF-728.001	0		18/04/2006	IF-728.001	01	-	2000
IF-780.001	0		20/04/2006	IF-780.001	01	-	3000
IF-729.001	0		18/04/2006	IF-729.001	01	-	3000
IF-786.008	0		20/04/2006	IF-786.008	01	-	5000
Codice parte : 0101119273302 (Quantita teorica SUM=2400)							
BF-66.014	0		08/04/2006	BF-66.014	01	-	2400

I subtotali possono essere inseriti dal menu contestuale accessibile cliccando col tasto destro sull'intestazione del campo sui cui valori si desidera calcolare il subtotali (nell'esempio il campo è "Quantità teorica") come visibile qui:



E' anche possibile di formattare a piacere i subtotali di gruppo, sia i subtotali dell'area footer del gruppo, sia i subtotali visualizzati nell'intestazione.

Il formato può essere modificato cliccando sulla nuova voce di menu "Display format" disponibile nel menu contestuale dei campi e nel menu contestuale dei group footer.

Codice ordine	Codice fase	Cc	Data disp	Codice commessa	Codice n		Quantita teorica		
Codice parte : 0101118320501 (La somma delle quantità teoriche è 17700)									
IF-794.001	0		20/04/2006	IF-794.001	01	-	200		V
IF-733.001	0		18/04/2006	IF-733.001	01	-	500		V
IF-672.002	0		11/04/2006	IF-672.002	01	-	1000		V
IF-781.002	0		20/04/2006	IF-781.002	01	-	1000		V
IF-730.001	0		18/04/2006	IF-730.001	01	-	2000		V
IF-728.001	0		18/04/2006	IF-728.001	01	-	2000		V
IF-780.001	0		20/04/2006	IF-780.001	01	-	3000		V
IF-729.001	0		18/04/2006	IF-729.001	01	-	3000		V
IF-786.008	0		20/04/2006	IF-786.008	01	-	5000		V
In questo gruppo ci sono 9 elementi							Il minimo è 200,00		
Codice parte : 0101119273302 (La somma delle quantità teoriche è 2400)									
BF-66.014	0		08/04/2006	BF-66.014	01	-	2400		V
In questo gruppo ci sono 1 elementi							Il minimo è 2.400,00		
Codice parte : 0101119273304 (La somma delle quantità teoriche è 2400)									
BF-93.001	0		19/04/2006	BF-93.001	01	-	2400		V
In questo gruppo ci sono 1 elementi							Il minimo è 2.400,00		

E' possibile ordinare le righe della griglia sulla base del valore di un sommario calcolato a livello di raggruppato.

Per attivare l'ordinamento è sufficiente cliccare col tasto destro sopra il sommario visualizzato nell'intestazione del gruppo e poi scegliere il tipo di ordinamento dal menu contestuale che verrà presentato.

Dallo stesso menu è possibile resettare questo tipo di ordinamento (scegliendo "Sort by Group Value").

Quando è attivato un ordinamento, il testo del sommario compare scritto in grassetto e di colore marrone.

8.6 Report pivot

Il report pivot è in grado di fornire una rappresentazione compatta di un set di dati molto esteso.

Pivot table su wikipedia

8.6.1 Introduzione pivot

L'abilità maggiore del report pivot è quella di permettere una riorganizzazione della rappresentazione spostando i campi all'interno di diverse aree o all'interno di particolari zone o nascondendoli sulla "customization form" associata alla pivot.

Attraverso il drag and drop dei campi la rappresentazione dei dati cambia al volo.

E' possibile modificare rapidamente l'orientamento di un campo da colonna da colonna a riga e arrangiare i campi all'interno dei una "header area" in modo che i valori di quei campi vengano organizzati in forma gerarchica. In questo modo l'informazione può essere organizzata per categorie ordinate come in un albero e l'utente può effettuare operazioni di drill down del dato ai diversi livelli di questo.

Il risultato di tutto ciò è che l'utente può svolgere un'analisi approfondita del dato ove necessario e mantenere nel contempo nascosta tutta l'informazione di dettaglio non necessaria.

E' anche possibile creare dei subtotali dei dati visualizzati.

Per snellire ulteriormente l'informazione visualizzata, l'utente può definire dei filtri.
I filtri possono essere applicati in due modi. Innanzitutto l'utente può effettuare il drag and drop di un campo nella "filter header area" e selezionare i valori da filtrare nel menu a tendina dell'intestazione del campo. Questo tipo di filtro ha effetto su tutti i dati visualizzati sull'intero report. In alternativa l'utente può filtrare i valori dei campi posizionati su altre aree oltre alla "filter header area".

Visualizzazione gerarchica

I valori dei campi sono visualizzati in modo gerarchico se due o più campi sono stati trascinati nell'header area del report. La vista gerarchica permette un'analisi a differenti livelli. Più è profonda la gerarchia, più livelli di analisi sono utilizzabili.
Nell'esempio qui sotto la tabella Orders contiene tra gli altri un campo "Company name" (il cliente) e un campo "Car Name". L'aver trascinato i 2 campi nella "Row header area" ha prodotto la seguente visualizzazione gerarchica:

Unit Price | PurchaseDate

Quantity | Payment Amount | PaymentType

Company Name	Car Name	Master Quantity	Master Payment Amount	Visa Quantity	Visa Payment Amount	Grand Total Quantity	Grand Total Payment Amount
⊟ Alessandro & Associates	BMW: 530i			2	$78,900.00	2	$78,900.00
	Cadillac: Seville	5	$248,000.00			5	$248,000.00
	Dodge: Ram 1500	1	$17,315.00	4	$69,260.00	5	$86,575.00
	Ford: Ranger FX-4			3	$37,695.00	3	$37,695.00
	GMC: Siera Quadrasteer	1	$17,748.00	7	$124,236.00	8	$141,984.00
	Jaguar: S-Type 3.0	12	$531,840.00			12	$531,840.00
	Lexus: GS 430			9	$371,178.00	9	$371,178.00
	Mercedes-Benz: C230 Kompressor Sport	4	$102,400.00			4	$102,400.00
	Mercedes-Benz: CLK55 AMG Cabriolet	2	$159,290.00			2	$159,290.00
	Mercedes-Benz: SL500 Roadster	5	$419,000.00	1	$83,800.00	6	$502,800.00
	Toyota: Tacoma S-Runner	4	$80,000.00	4	$80,000.00	8	$160,000.00
Alessandro & Associates Total		34	$1,575,593.00	30	$845,069.00	64	$2,420,662.00
⊞ Christies House of Design		46	$2,739,352.00	62	$5,416,125.00	108	$8,155,477.00

Nell'esempio si può vedere come i campi siano sommarizzati per cliente e poi per modello. I valori di cliente mostrano un pulsante per espandere o comprimere il nodo della gerarchia. In questo modo l'utente può passare da una vista "per cliente" a quella più dettagliata "per modello".

8.6.2 Struttura pivot

Il report pivot è organizzato in aree.

L'**Header area** mostra le intestazioni di tutti i campi visibili. Questa area è composta dalla Column header area, dalla Data header area, dalla Filter header area e dalla Row header area.

Filter Header Area
Data Header Area
Row Header Area
Header Area
Column Header Area

Purchase Date

Qty | Payment Amount | Month | Data

Payment Type	Car Name	January 2002		February 2002		Grand Total	
		Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
AmEx	Cadillac: DeVille			10	$477,800.00	10	$477,800.00
	BMW: 530i	2	$78,900.00	14	$552,300.00	16	$631,200.00
AmEx Total		2	$78,900.00	24	$1,030,100.00	26	$1,109,000.00
Master	BMW: 530i			8	$315,600.00	8	$315,600.00
	Cadillac: DeVille	2	$95,560.00	5	$238,900.00	7	$334,460.00
Master Total		2	$95,560.00	13	$554,500.00	15	$650,060.00
Visa	Cadillac: DeVille	1	$47,780.00			1	$47,780.00
	BMW: 530i			12	$473,400.00	12	$473,400.00
Visa Total		1	$47,780.00	12	$473,400.00	13	$521,180.00
Grand Total		5	$222,240.00	49	$2,058,000.00	54	$2,280,240.00

La **Data header area** contiene le intestazioni di tutti i campi i cui dati, subtotali e grand total sono mostrati nella "Data area".

La **Column header area** contiene le intestazioni di tutte i campi colonna e la "Data field anchor" (quando presente).

La **Row header area** contiene le intestazioni di tutti i campi riga e il "Data field anchor" (quando presente).

La **Filter header area** contiene le instestazioni di tutti i campi filtro (il cui valore viene utilizzato per filtrare i dati visualizzati nella pivot

ma senza essere sommarizzato come i valori dei campi della Data header area).

8.6.2.1 Data area e data header area

La Data area contiene i valori aggregati.

La Data header area contiene tutti i campi data il cui valore è aggregato nella Data area della pivot.

Payment Type	Car Name	January 2002		February 2002		Grand Total	
		Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
AmEx	Cadillac: DeVille			10	$477,800.00	10	$477,800.00
	BMW: 530i	2	$78,900.00	14	$552,300.00	16	$631,200.00
AmEx Total		2	$78,900.00	24	$1,030,100.00	26	$1,109,000.00
Master	BMW: 530i			8	$315,600.00	8	$315,600.00
	Cadillac: DeVille	2	$95,560.00	5	$238,900.00	7	$334,460.00
Master Total		2	$95,560.00	13	$554,500.00	15	$650,060.00
Visa	Cadillac: DeVille	1	$47,780.00			1	$47,780.00
	BMW: 530i			12	$473,400.00	12	$473,400.00
Visa Total		1	$47,780.00	12	$473,400.00	13	$521,180.00
Grand Total		5	$222,240.00	49	$2,058,000.00	54	$2,280,240.00

Nel menu contestuale dei campi presenti in questa area sono invocabili i comandi:

- Hide
- Show field list
- Tipo sommarizzazione
- Abilita gruppo

8.6.2.2 Column header area

Nella Column Header Area sono visibili tutti i campi colonna e il Data Field Anchor (se presente).

Filter Header Area
Header Area
Data Header Area
Column Header Area
Row Header Area
Column Area

Purchase Date

Qty | Payment Amount | Month | Data

Payment Type	Car Name	January 2002		February 2002		Grand Total	
		Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
AmEx	Cadillac: DeVille			10	$477,800.00	10	$477,800.00
	BMW: 530i	2	$78,900.00	14	$552,300.00	16	$631,200.00
AmEx Total		2	$78,900.00	24	$1,030,100.00	26	$1,109,000.00
Master	BMW: 530i			8	$315,600.00	8	$315,600.00
	Cadillac: DeVille	2	$95,560.00	5	$238,900.00	7	$334,460.00
Master Total		2	$95,560.00	13	$554,500.00	15	$650,060.00
Visa	Cadillac: DeVille	1	$47,780.00			1	$47,780.00
	BMW: 530i			12	$473,400.00	12	$473,400.00
Visa Total		1	$47,780.00	12	$473,400.00	13	$521,180.00
Grand Total		5	$222,240.00	49	$2,058,000.00	54	$2,280,240.00

Nel menu contestuale di questa area sono invocabili i comandi:

- Hide
- Show field list
- Tipo Raggruppamento
- Totali per Sommarizzazioni
- Posizione Totali
- Ordina su Grand Total
- Mostra solo
- Trova

8.6.2.3 Row header area

Nella Row header area sono visibili tutti i campi riga e il Data field anchor (se presente). Le intestazioni dei campi riga sono visualizzati nella Row area.

Filter Header Area
Header Area
Purchase Date
Data Header Area
Column Header Area
Qty | Payment Amount
Month | Data
Row Header Area
Payment Type | Car Name
Row Area

Payment Type	Car Name	January 2002		February 2002		Grand Total	
		Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
AmEx	Cadillac: DeVille			10	$477,800.00	10	$477,800.00
	BMW: 530i	2	$78,900.00	14	$552,300.00	16	$631,200.00
AmEx Total		2	$78,900.00	24	$1,030,100.00	26	$1,109,000.00
Master	BMW: 530i			8	$315,600.00	8	$315,600.00
	Cadillac: DeVille	2	$95,560.00	5	$238,900.00	7	$334,460.00
Master Total		2	$95,560.00	13	$554,500.00	15	$650,060.00
Visa	Cadillac: DeVille	1	$47,780.00			1	$47,780.00
	BMW: 530i			12	$473,400.00	12	$473,400.00
Visa Total		1	$47,780.00	12	$473,400.00	13	$521,180.00
Grand Total		5	$222,240.00	49	$2,058,000.00	54	$2,280,240.00

Nel menu contestuale di questa area sono invocabili i comandi:

- Hide
- Show field list
- Tipo Raggruppamento
- Totali per Sommarizzazioni
- Posizione Totali
- Ordina su Grand Total
- Mostra solo
- Trova

8.6.2.4 Filter header area

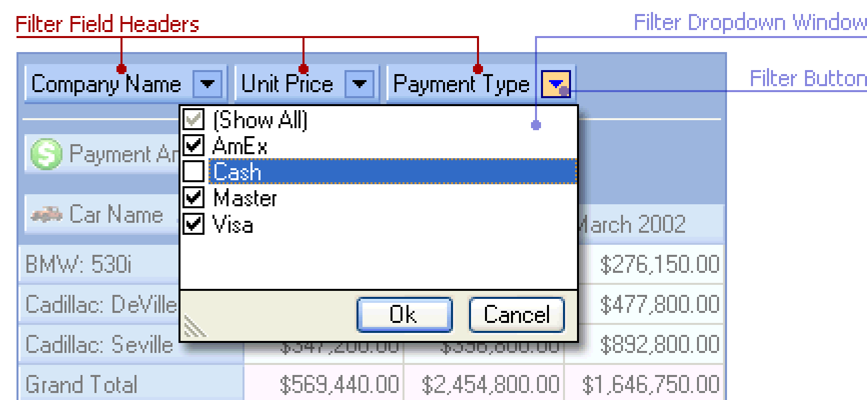
I campi filtro vengono posizionati nella filter header area. Questi campi permettono all'utente di applicare un filtro su tutta la pivot in modo da visualizzare i dati relativi solo a specifici valori. Il filtro può essere impostato dalla finestra a tendina che è invocata cliccando sul "Filter button".

Quando un filtro è attivo, il Filter button viene colorato di blu.

Filter Header Area
Purchase Date
Header Area
Data Header Area
Qty
Payment Amount
Month
Data
Column Header Area
Row Header Area
Payment Type
Car Name

Payment Type	Car Name	January 2002		February 2002		Grand Total	
		Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
AmEx	Cadillac: DeVille			10	$477,800.00	10	$477,800.00
	BMW: 530i	2	$78,900.00	14	$552,300.00	16	$631,200.00
AmEx Total		2	$78,900.00	24	$1,030,100.00	26	$1,109,000.00
Master	BMW: 530i			8	$315,600.00	8	$315,600.00
	Cadillac: DeVille	2	$95,560.00	5	$238,900.00	7	$334,460.00
Master Total		2	$95,560.00	13	$554,500.00	15	$650,060.00
Visa	Cadillac: DeVille	1	$47,780.00			1	$47,780.00
	BMW: 530i			12	$473,400.00	12	$473,400.00
Visa Total		1	$47,780.00	12	$473,400.00	13	$521,180.00
Grand Total		5	$222,240.00	49	$2,058,000.00	54	$2,280,240.00

Per aggiungere altre condizione di filtraggio dei dati, l'utente può anche applicare dei criteri di filtro sui campi presenti nella Column header area e nella Row header area.



Nel menu contestuale dei campi presenti in questa area sono invocabili i comandi:

- Hide
- Show field list

8.6.2.5 Data field anchor

In caso di utilizzo di campi data multipli, la pivot mette a disposizione un campo speciale chiamato Data Field Anchor. Questo campo agisce come un campo separato a cui gli altri campi data vengono collegati automaticamente. Il data field anchor è utile nel caso l'utente voglia riposizionare tutti i campi data in un colpo solo. Draggando il data field anchor in una particolare posizione si otterrà di spostare i campi data in quella posizione.

Nell'esempio il data field anchor è posizione tra "Year" e "Quarter" e quindi nella column area viene visualizzato prima "Year", poi i 2 campi

data "Qty" e "Payment Amount" e poi "Quarter".

Data Fields · Data Header Area · Data Field Anchor

Unit Price

Qty | Payment Amount | Year | Data | Quarter

Car Name	2002 Qty Qtr 1	2002 Qty Qtr 2	2002 Payment Amount Qtr 1	2002 Payment Amount Qtr 2	2002 Total Qty	2002 Total Payment Amount
BMW: 530i	57	59	$2,248,650.00	$2,327,550.00	116	$4,576,200.00
Cadillac: DeVille	39	52	$1,863,420.00	$2,484,560.00	91	$4,347,980.00
Cadillac: Seville	53	61	$2,628,800.00	$3,025,600.00	114	$5,654,400.00
Grand Total	149	172	$6,740,870.00	$7,837,710.00	321	$14,578,580.00

Data Area · Data Field Headers

Visto che il data field anchor rappresenta un insieme di campi data, può essere posizionato solo nella column header area o nella row header area.

Data Fields

Data Header Area

Unit Price

Qty | Payment Amount

Year | Quarter

Data Field Anchor

Data	Car Name	2002 Qtr 1	2002 Qtr 2	2002 Total
Qty	BMW: 530i	57	59	116
	Cadillac: DeVille	39	52	91
	Cadillac: Seville	53	61	114
Payment Amount	BMW: 530i	$2,248,650.00	$2,327,550.00	$4,576,200.00
	Cadillac: DeVille	$1,863,420.00	$2,484,560.00	$4,347,980.00
	Cadillac: Seville	$2,628,800.00	$3,025,600.00	$5,654,400.00
Grand Total	Qty	149	172	321
	Payment Amount	$6,740,870.00	$7,837,710.00	$14,578,580.00

Data Field Headers

Data Area

Nell'esempio il data field anchor è posizionato nella row header area, prima del campo "Car Name". La Row area rispecchia questa configurazione.

8.6.3 Totali e subtotali pivot

Subtotali

I subtotali sono calcolati e visualizzati per i campi i cui valori sono visualizzati nella pivot. Nell'esempio il campo data è "Payment Amount" ed il subtotale è calcolato per ogni livello di aggregazione dei dati.

Total for the Qtr 1 Column

Total for the 2002 Column

Payment Amount		Year / Quarter / Month					
		2002					2002 Total
		Qtr 1			Qtr 1 Total	Qtr 2	
Payment Type	Car Name	January	February	March		April	
AmEx	BMW: 530i	$78,900.00	$552,300.00	$276,150.00	$907,350.00		$907,350.00
	Cadillac: DeVille		$477,800.00	$47,780.00	$525,580.00	$191,120.00	$716,700.00
AmEx Total		$78,900.00	$1,030,100.00	$323,930.00	$1,432,930.00	$191,120.00	$1,624,050.00
Master		$95,560.00	$554,500.00	$430,020.00	$1,080,080.00	$579,490.00	$1,659,570.00
Visa	BMW: 530i		$473,400.00		$473,400.00	$157,800.00	$631,200.00
	Cadillac: DeVille	$47,780.00			$47,780.00	$238,900.00	$286,680.00
Visa Total		$47,780.00	$473,400.00		$521,180.00	$396,700.00	$917,880.00
Grand Total		$222,240.00	$2,058,000.00	$753,950.00	$3,034,190.00	$1,167,310.00	$4,201,500.00

Total for the Visa Row

Total for the AmEx Row

Un subtotale è calcolato per ogni campo data. Di default, se sono presenti più campi nella visualizzazione, la pivot mostra diverse colonne di totali e una singola riga di totale per ogni campo. Nell'immagine qui sotto la pivot contiene 2 campi data ("Payment Amount" e "Qty") e mostra 2 colonne di subtotali (una per "Payment Amount" e una per "Qty") e una singola riga di totali (che ospita i valori sia di "Payment Amount" sia di "Qty") per tutti i valori dei 2 campi.

Total for the Qtr 1 Column

Total for the 2002 Column

Payment Amount	Qty	Year / Quarter / Month											
		2002										2002 Total	
		Qtr 1						Qtr 1 Total		Qtr 2			
		January		February		March				April			
Payment Type	Car Name	Payment Amount	Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount	Qty
AmEx	BMW: 530i	$78,900.00	2	$552,300.00	14	$276,150.00	7	$907,350.00	23			$907,350.00	23
	Cadillac: DeVille			$477,800.00	10	$47,780.00	1	$525,580.00	11	$191,120.00	4	$716,700.00	15
AmEx Total		$78,900.00	2	$1,030,100.00	24	$323,930.00	8	$1,432,930.00	34	$191,120.00	4	$1,624,050.00	38
Master		$95,560.00	2	$554,500.00	13	$430,020.00	9	$1,080,080.00	24	$579,490.00	13	$1,659,570.00	37
Visa	BMW: 530i			$473,400.00	12			$473,400.00	12	$157,800.00	4	$631,200.00	16
	Cadillac: DeVille	$47,780.00	1					$47,780.00	1	$238,900.00	5	$286,680.00	6
Visa Total		$47,780.00	1	$473,400.00	12			$521,180.00	13	$396,700.00	9	$917,880.00	22
Grand Total		$222,240.00	5	$2,058,000.00	49	$753,950.00	17	$3,034,190.00	71	$1,167,310.00	26	$4,201,500.00	97

Total for the Visa Row

Total for the AmEx Row

E' possibile ribaltare il layout e mostrare subtotali per riga multipli e una sola colonna di subtotali spostando col drag and drop la "data field anchor" nella "row header area".

Le impostazioni dei subtotali possono essere variate attraverso il menu contestuale accessibile cliccando col tasto destro sul campo, la voce di menu è "Totali per Sommarizzazioni":

I subtotali sono calcolati in questo modo:

Value	Meaning
stCount	Counts values
stSum	Provides a total of numeric values $\mathbf{Sum} = \sum_{i}^{n} v_i$
stMin	Provides the lowest value
stMax	Provides the highest value
stAverage	Provides an average of numeric values $\bar{v} = \frac{1}{n}\sum_{i} v_i$
stStdDev	Provides the sample standard deviation of numeric values $\mathbf{StdDev} = \sqrt{\frac{1}{n-1}\sum_{i}(v_i - \bar{v})^2}$
stStdDevP	Provides the population standard deviation of numeric values $\mathbf{StdDevp} = \sqrt{\frac{1}{n}\sum_{i}(v_i - \bar{v})^2}$
stVariance	Provides the sample variance of numeric values $\mathbf{Var} = \frac{1}{n-1}\sum_{i}(v_i - \bar{v})^2$
stVarianceP	Provides the population variance of numeric values $\mathbf{Varp} = \frac{1}{n}\sum_{i}(v_i - \bar{v})^2$
stCustom	Provides the custom calculation

Grand-total

I grand total sono calcolati automaticamente per tutti i dati visualizzati all'interno della pivot.

I grand total orizzontali sono righe che mostrano i totali calcolati sui valori di tutte le righe, così come i grand total verticali sono colonne che mostrano i totali calcolati sui valori di tutte le colonne.

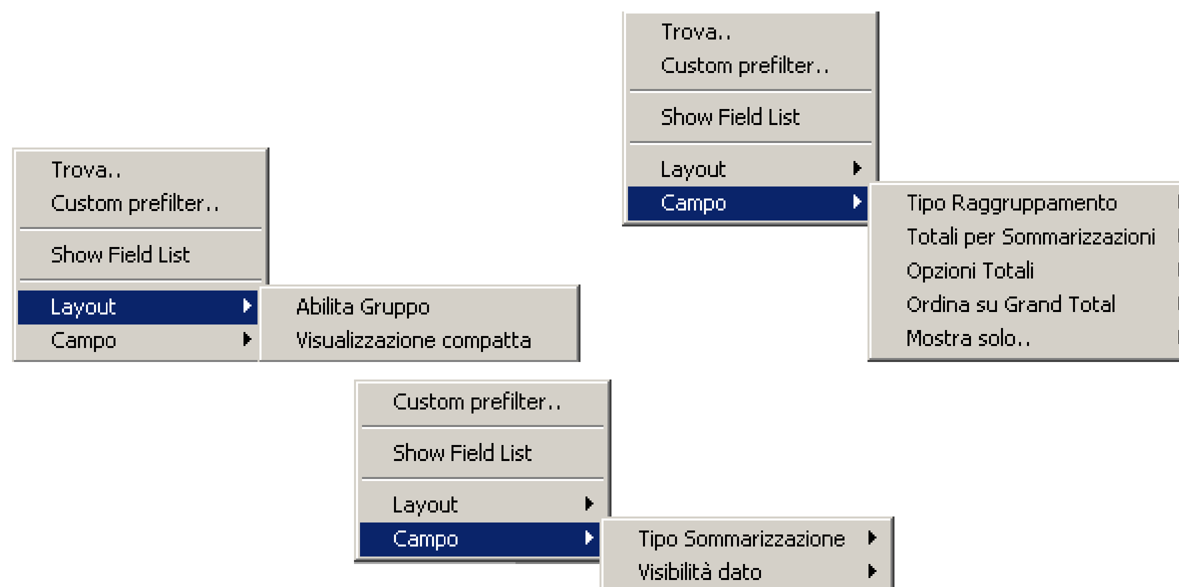
Column Grand Total

Payment Amount		Year / Quarter / Month					
		2002				2002 Total	Grand Total
		Qtr 1		Qtr 1 Total	Qtr 2		
Payment Type	Car Name	January	February				
AmEx	BMW: 530i	$78,900.00	$552,300.00	$631,200.00		$631,200.00	$631,200.00
	Cadillac: DeVille		$477,800.00	$477,800.00	$430,020.00	$907,820.00	$907,820.00
AmEx Total		$78,900.00	$1,030,100.00	$1,109,000.00	$430,020.00	$1,539,020.00	$1,539,020.00
Visa	BMW: 530i		$473,400.00	$473,400.00	$157,800.00	$631,200.00	$631,200.00
	Cadillac: DeVille	$47,780.00		$47,780.00	$382,240.00	$430,020.00	$430,020.00
Visa Total		$47,780.00	$473,400.00	$521,180.00	$540,040.00	$1,061,220.00	$1,061,220.00
Grand Total		$126,680.00	$1,503,500.00	$1,630,180.00	$970,060.00	$2,600,240.00	$2,600,240.00

Row Grand Total

8.6.4 Menu contestuale standard pivot

Cliccando col tasto destro sull' Header area della pivot è disponibile un menu contestuale che contiene diversi comandi variabili a seconda di quale elemento dell'area si trova sotto il cursore del mouse.



Cliccando col tasto destro sulla Column area o sulla Row area è disponibile un menu contestuale che contiene alcuni comandi per collassare o espandere i nodi della gerarchia. E' possibile anche copiare negli appunti il contenuto della cella di intestazione su cui si è cliccato col mouse. Il menu viene reso visibile solo in presenza di più campi nelle corrispettive aree header.



8.6.4.1 Menu Trova

Ricerca un valore in un campo scelto dall'utente a partire dalla cella attualmente selezionata.

Se la cella su cui è invocato il comando appartiene alla Row header area la direzione di ricerca è dall'alto al basso, se la cella su cui è invocato il comando appartiene alla Column header area la direzione di ricerca è da sinistra verso destra.

8.6.4.2 Menu Custom Prefilter

Crea un prefiltro al volo partendo dalla colonna selezionata utilizzando una finestra di inserimento semplificata:

8.6.4.3 Menu Hide

Nasconde il campo su cui è stato invocato il menu contestuale. Il campo può essere recuperato nella Customization Form.

8.6.4.4 Menu Show Field List

Visualizza la Customization Form.

8.6.4.5 Menu Abilita Gruppo

I field della Data header area possono essere raggruppati insieme in modo da formare una catena indivisibile.

Tempo lav simulato - Formula1 - Formula2 - Valenza Minima | Data ora inizio simulata | Data

Numero fase	Codice risorsa	14/07/2008 Tempo lav simulato	Formula1	Formula2	Valenza Minima	21/07/20 Tempo lav
5	CONF-1KG	8,00	16,00	4,00	2,00	

I campi raggruppati in una catena vengono mossi tutti assieme quando oggetto di drag and drop.

La catena può essere avvolta o svolta utilizzando i pulsanti di implosione/esplosione visualizzati su tutti i campi della catena. Man mano che la catena viene avvolta, i campi vengono nascosti così come i loro valori. Se la catena viene svolta, i campi e i corrispettivi valori ritornano visibili man mano che si procede lungo la catena.

Tempo lav simulato - Formula1 | Data ora inizio simulata | Data

Numero fase	Codice risorsa	14/07/2008 Tempo lav simulato	Formula1	21/07/2008 Tempo lav sim
5	CONF-1KG	8,00	16,00	

Questo meccanismo può aiutare a migliorare la leggibilità del report.

8.6.4.6 Menu Visualizzazione Compatta

Abilita o disabilita il layout "compatto" della pivot.

La visualizzazione compatta è utile per recuperare spazio orizzontalmente senza sacrificare la distinzione dei valori presenti nella gerarchia della pivot.

Sotto una pivot con la visualizzazione normale e più in basso la stessa pivot con la visualizzazione compatta attivata:

Risorsa prevista

Valenza simulata | Ore primo giorno | Ore ultimo giorno | Data ora inizio simulata

Codice ordine	Numero fase	Codice risorsa	Valenza simulata	Or
200600027739	10	ESSICCATORE	001	
	10 Total		001	
200600027739 Sum			1,00	
200600027739 Min			1,00	
200600027739 Max			1,00	
200600028772	30	ESSICCATORE	001	
	30 Total		001	
200600028772 Sum			1,00	
200600028772 Min			1,00	

Risorsa prevista

Valenza sin | Ore primo (| Ore ultimo (| Data ora inizio simulata | D

Codice | Numero | Codice

	Grand	
	Valenza simulata	Ore primo
200600027739	001	
10	001	
ESSICCATORE	001	
200600027739 Sum	1,00	
200600027739 Min	1,00	
200600027739 Max	1,00	
200600028772	001	
30	001	
ESSICCATORE	001	
200600028772 Sum	1,00	
200600028772 Min	1,00	

8.6.4.7 Menu Tipo Sommarizzazione

Invocato su un campo della Data header area, questo comando permette di cambiare il tipo di sommarizzazione (somma, media, min, max, ecc..) che la pivot effettua sui valori del campo.
Il default è Sum (somma): i valori del campo vengono sommati tra di loro dopo essere stati raggruppati secondo la gerarchia impostata dalla configurazione dei campi colonna e riga.

Sui campi provenienti da formule su aggregato il tipo sommarizzazione non può essere impostato (non avrebbe senso).

8.6.4.8 Menu Tipo Raggruppamento

La pivot raggruppa i valori dei campi data (della Data header area) secondo il Tipo Raggruppamento definito sui campi colonna (della Column header area) e riga (della Row header area) e poi calcola la sommarizzazione secondo il Tipo Sommarizzazione definito per i

campi data (dalla Data header area).

Se il campo colonna o riga è di tipo data/ora è possibile definire un tipo di raggruppamento tra quelli predefiniti. Ad esempio:

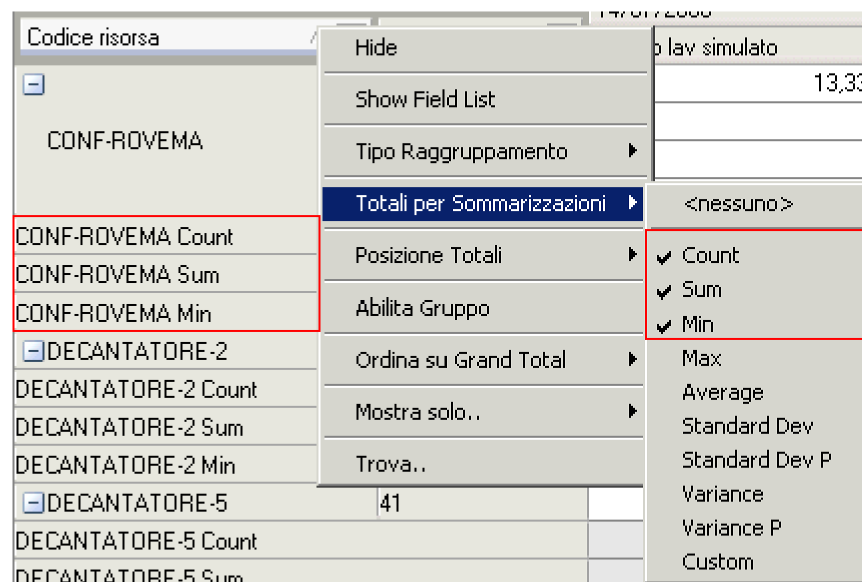
- Default (nessun raggruppamento)
- Mese (raggruppa tutti i valori di Gennaio 2008, poi Febbraio 2008, Marzo 2008, ecc..)
- Anno (raggruppa tutti i valori di 2008, poi di 2009, ecc..)
- Giorno assoluto (raggruppa tutti i valori dei giorni 1, poi 2, poi 3, ecc..)
- Mese assoluto (raggruppa tutti i valori di Gennaio, di Febbraio, ecc...)
- ecc..

Se il campo colonna o riga è di tipo stringa è possibile scegliere un raggruppamento alfabetico. Vengono raggruppati tutti i dati associati a campi colonna o riga che iniziano per A, poi per B, poi per C, ecc...

Se il campo colonna o riga è di tipo numero è possibile scegliere un raggruppamento numerico. Vengono raggruppati tutti i dati associati a valori del campo colonna o riga compreso tra 0 e 9, poi quelli associati a valori tra 10 e 19, ecc..

8.6.4.9 Menu Totali per Sommarizzazioni

Aggiunge uno o più subtotali al campo corrente.



Il tipo "Custom" serve ad aggiungere un subtotale (calcolato sempre come Sum) per i campi derivati da Formule su aggregato.

Ulteriori informazioni sul calcolo delle sommarizzazioni sono disponibili nella sezione Totali e subtotali pivot.

8.6.4.10 Menu Opzioni Totali

Questo menu permette di modificare alcune opzioni riguardo il calcolo e la rappresentazione dei totali.

8.6.4.10.1 Menu Visibilità Totali

Permette di specificare se si desidera visualizzare i grand total per riga e per colonna o se si desidera mantenerli nascosti.

8.6.4.10.2 Menu Posizione Totali Colonne

Invocato su un elemento della row header area o della column header area, questo comando sposta il grand total calcolato

orizzontalmente all'inizio o alla fine della pivot (a sinistra o a destra).

8.6.4.10.3 Menu Base di calcolo Totali

Permette di specificare come i valori dei totali devono essere calcolati a partire dai dati delle celle della pivot.

- Usa tutti i dati (default)

I totali sono calcolati utilizzando tutti i dati corrispondenti alla cella del totale. Questo è il metodo di calcolo di default.

Qty / Payment Amount		Company Name / Payment Type					
		Doe Enterprises		Hill Corporation		Grand Total	
Month	Car Name	Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
February 2002	Dodge: Ram 1500			7	$121,205.00	7	$121,205.00
	BMW: 530i	4	$157,800.00	1	$39,450.00	5	$197,250.00
	Cadillac: DeVille	5	$238,900.00	4	$191,120.00	9	$430,020.00
February 2002 Average		3.00	$132,233.33	3.00	$87,943.75	3.00	$106,925.00
February 2002 Sum		9	$396,700.00	12	$351,775.00	21	$748,475.00
February 2002 Min		1	$47,780.00	1	$39,450.00	1	$39,450.00
February 2002 Max		4	$191,120.00	4	$191,120.00	4	$191,120.00
February 2002 Count		3	3	4	4	7	7

- Usa i dati visibili del livello superiore

I totali sono calcolati usando i dati visibili al livello di dettaglio immediatamente superiore.

Qty / Payment Amount		Company Name / Payment Type					
		Doe Enterprises		Hill Corporation		Grand Total	
Month	Car Name	Qty	Payment Amount	Qty	Payment Amount	Qty	Payment Amount
February 2002	Dodge: Ram 1500			7	$121,205.00	7	$121,205.00
	BMW: 530i	4	$157,800.00	1	$39,450.00	5	$197,250.00
	Cadillac: DeVille	5	$238,900.00	4	$191,120.00	9	$430,020.00
February 2002 Average		4.50	$198,350.00	4.00	$117,258.33	4.20	$149,695.00
February 2002 Sum		9	$396,700.00	12	$351,775.00	21	$748,475.00
February 2002 Min		4	$157,800.00	1	$39,450.00	1	$39,450.00
February 2002 Max		5	$238,900.00	7	$191,120.00	7	$238,900.00
February 2002 Count		2	2	3	3	5	5

8.6.4.10.4 Menu Valori nulli Totali

Specifica se i valori nulli devono essere ignorati quando il calcolo dei totali viene effettuato.

Di default i valori nulli vengono computati durante il calcolo dei totali.

Questa opzione è particolarmente importante per il calcolo dei valori medi.

8.6.4.10.5 Menu Valori unici Totali

Permette di definire la visibilità dei grand total e dei subtotali in presenza di valori unici.

Di default se è presente una sola riga/colonna di dati aggregati, il corrispondente grand total non è visualizzato (i valori sarebbero un duplicato dell'unica riga/colonna presente).

Discorso invertito per i subtotali: anche in presenza di una unica riga/colonna, i subtotali intermedi sono sempre visualizzati.

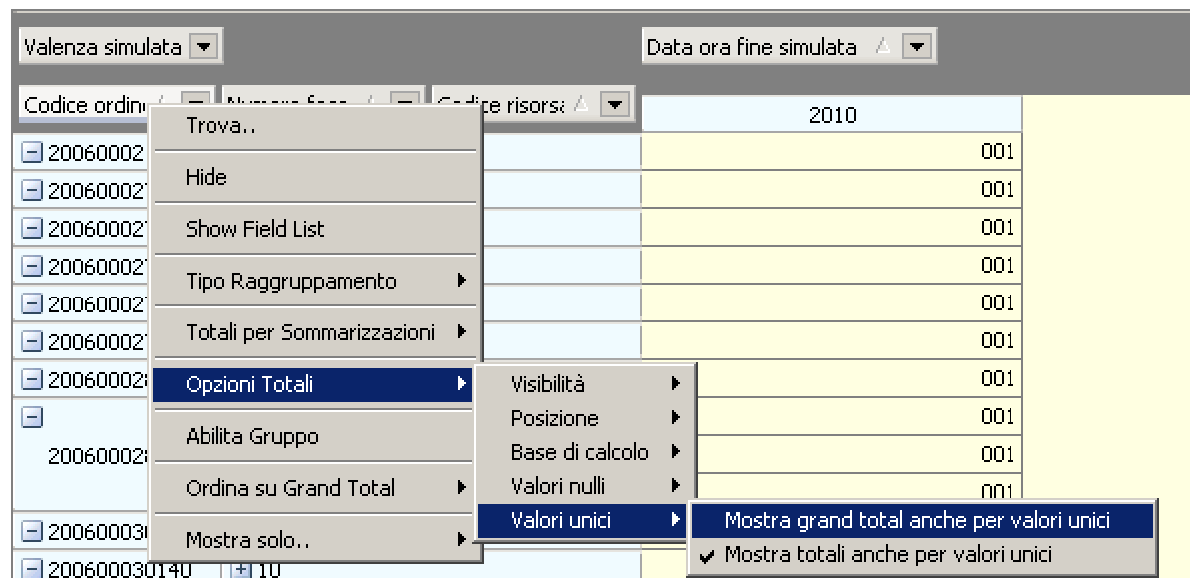
Attraverso queste opzioni è possibile modificare il comportamento di default e decidere se visualizzare grand total e subtotali in presenza di valori unici.

Subtotali visibili anche per valori unici:

Subtotali nascosti per valori unici:

Grand total visibili anche per valori unici:

Grand total nascosti per valori unici:



8.6.4.11 Menu Visibilità Dato

Se sono presenti **almeno due field** nella Data header area, è possibile modificare la visibilità degli stessi nella Data area agendo su questo menu.

Per ogni field è infatti possibile decidere se renderne visibili i valori solo come celle dei valori ("Celle cross e total"), solo come grand total ("Celle grand total") o entrambi ("Tutte le celle"), che è il default.

Visibilità "Tutte le celle" per il campo Tempo lavorazione simulato:

Valenza sim | Tempo lav sim | Data ora fine simulata | Data

Codice risorsa	mar 2009		apr 2009		mag 2009		gen 2010		Grand Total	
	Valenza si	Tempo la	Valenza si	Tempo la	Valenza si	Tempo la	Valenza si	Tempo la	Valenza si	Tempo la
CDL10	1	4,80							1,00	4,80
CONF-1KG	3	16,00							3,00	16,00
CONF-GRANULARI	6	93,14							6,00	93,14
CONF-MANUALE	14	89,30	1	4,00	1	500,00			16,00	593,30
CONF-ROVEMA	16	126,53	29	115,87					45,00	242,40
DECANTATORE-2	1	206,45							1,00	206,45
DECANTATORE-5	1	0,05							1,00	0,05
ESSICCATORE	8	281,42	6	287,03					14,00	568,45
ESTRATTORE							1	6088,00	1,00	6088,00
FILTRATRICE-1	3	10,62							3,00	10,62
IDROSOLUBILI	22	135,53	13	115,54					35,00	251,07
LIQCONF1L-1KG	19	71,62							19,00	71,62
LIQCONFAUT1L-5L	31	74,80	14	34,05					45,00	108,85
LIQCONFMAN	11	163,52							11,00	163,52
MISC-FOGLIARI-1	8	60,64	3	24,83					11,00	85,46
MISC-FOGLIARI-2	11	49,93	3	9,69					14,00	59,62
PELLET	2	18,85							2,00	18,85
REATTORE-1	11	26,20							11,00	26,20
REATTORE-2	10	39,14							10,00	39,14
REATTORE-3							1	'0000,00	1,00	'0000,00
REATTORE-4	8	34,29							8,00	34,29
REATTORE-5	8	34,49							8,00	34,49
REATTORECHEL-01	2	66,72	1	46,67					3,00	113,38
REATTORECHEL-02	1	200,00							1,00	200,00
REATTORECHEL-03	1	0,05	1	700,00					2,00	700,05
SISTEMA-HA			6	272,05					6,00	272,05
Grand Total	198,00	1804,09	77,00	1609,72	1,00	500,00	2,00	6088,00	278,00	'0001,80

Visibilità "Celle cross e total" per il campo Tempo lavorazione simulato:

Valenza sim	Tempo lav sim	Data ora fine simulata	Data						
	mar 2009		apr 2009		mag 2009		gen 2010		Grand Tot
Codice risorsa	Valenza si	Tempo la	Valenza si	Tempo la	Valenza si	Tempo la	Valenza si	Tempo la	Valenza si
CDL10	1	4,80							1,00
CONF-1KG	3	16,00							3,00
CONF-GRANULARI	6	93,14							6,00
CONF-MANUALE	14	89,30	1	4,00	1	500,00			16,00
CONF-ROVEMA	16	126,53	29	115,87					45,00
DECANTATORE-2	1	206,45							1,00
DECANTATORE-5	1	0,05							1,00
ESSICCATORE	8	281,42	6	287,03					14,00
ESTRATTORE							1	6088,00	1,00
FILTRATRICE-1	3	10,62							3,00
IDROSOLUBILI	22	135,53	13	115,54					35,00
LIQCONF1L-1KG	19	71,62							19,00
LIQCONFAUT1L-5L	31	74,80	14	34,05					45,00
LIQCONFMAN	11	163,52							11,00
MISC-FOGLIARI-1	8	60,64	3	24,83					11,00
MISC-FOGLIARI-2	11	49,93	3	9,69					14,00
PELLET	2	18,85							2,00
REATTORE-1	11	26,20							11,00
REATTORE-2	10	39,14							10,00
REATTORE-3							1	'0000,00	1,00
REATTORE-4	8	34,29							8,00
REATTORE-5	8	34,49							8,00
REATTORECHEL-01	2	66,72	1	46,67					3,00
REATTORECHEL-02	1	200,00							1,00
REATTORECHEL-03	1	0,05	1	700,00					2,00
SISTEMA-HA			6	272,05					6,00
Grand Total	198,00	1804,09	77,00	1609,72	1,00	500,00	2,00	i6088,00	278,00

Visibilità "Celle grand total" per il campo Tempo lavorazione simulato:

Valenza sim | Tempo lav sim | Data ora fine simulata | Data

Codice risorsa	mar 2009 Valenza si	apr 2009 Valenza si	mag 2009 Valenza si	gen 2010 Valenza si	Grand Total Valenza si	Tempo la
CDL10	1				1,00	4,80
CONF-1KG	3				3,00	16,00
CONF-GRANULARI	6				6,00	93,14
CONF-MANUALE	14	1	1		16,00	593,30
CONF-ROVEMA	16	29			45,00	242,40
DECANTATORE-2	1				1,00	206,45
DECANTATORE-5	1				1,00	0,05
ESSICCATORE	8	6			14,00	568,45
ESTRATTORE				1	1,00	6088,00
FILTRATRICE-1	3				3,00	10,62
IDROSOLUBILI	22	13			35,00	251,07
LIQCONF1L-1KG	19				19,00	71,62
LIQCONFAUT1L-5L	31	14			45,00	108,85
LIQCONFMAN	11				11,00	163,52
MISC-FOGLIARI-1	8	3			11,00	85,46
MISC-FOGLIARI-2	11	3			14,00	59,62
PELLET	2				2,00	18,85
REATTORE-1	11				11,00	26,20
REATTORE-2	10				10,00	39,14
REATTORE-3				1	1,00	'0000,00
REATTORE-4	8				8,00	34,29
REATTORE-5	8				8,00	34,49
REATTORECHEL-01	2	1			3,00	113,38
REATTORECHEL-02	1				1,00	200,00
REATTORECHEL-03	1	1			2,00	700,05
SISTEMA-HA		6			6,00	272,05
Grand Total	198,00	77,00	1,00	2,00	278,00	'0001,80

8.6.4.12 Menu Ordina su Grand Total

Ordina i valori di un campo secondo il valore del grand total di uno dei campi della Data header area.

Nell'esempio le righe del campo "Codice risorsa" sono ordinate secondo il grand total calcolato come somma (Sum) del campo "Tempo lav simulato".

Quando un campo è ordinato per i valori di grand total di un campo data, nell'intestazione del campo compare la scritta "[Gt]".

Tempo lav simulato

Data ora inizio simulata

Codice risorsa [Gt]

	lug 2008	ago 2008	Grand Total
DE	,05		,05
FI	13,62		13,62
CO	14,67		14,67
PE	14,79		14,79
RE	20,32		20,32
RE	20,95		20,95
RE	22,55		22,55
RE			23,40
RE			27,20
CO			32,92
MISC-FOGLIAR		1,92	56,06
MISC-FOGLIARI-1			87,11
CONF-MANUALE		4,00	89,30
LIQCONFAUT1L-5L			112,69
REATTORECHEL-01		6,67	113,38
LIQCONFMAN		0,24	142,56
CONF-ROVEMA	175,36	22,96	198,32
REATTORECHEL-02	200,00		200,00
DECANTATORE-2	206,45		206,45
IDROSOLUBILI	222,95		222,95
SISTEMA-HA		272,05	272,05
ESSICCATORE	352,40	216,05	568,45
REATTORECHEL-03	700,05		700,05
ESTRATTORE	10.716,00		10.716,00
Grand Total	13.291,96	583,88	13.875,84

Hide
Show Field List
Tipo Raggruppamento
Totali per Sommarizzazioni
Posizione Totali
Ordina su Grand Total
Mostra solo..
Trova..

Tempo lav simulato

Count
Sum
Min
Max
Average
Standard Dev
Standard Dev P
Variance
Variance P
Custom

8.6.4.13 Menu Mostra solo

Qualche volta l'utente non è interessato a visualizzare il report intero ma ha solo bisogno di vedere i valori migliori o peggiori. Per questo scopo è disponibile il comando "Mostra solo" che serve a nascondere una parte di valori dalla visualizzazione e a far vedere solo i migliori o peggiori valori in accordo col criterio di ordinamento attualmente definito.

Ad esempio partendo da questa pivot completa:

Codice risorsa	lug 2008	ago 2008	Grand Total
CONF-1KG	3,00		3,00
CONF-GRANULARI	3,00		3,00
CONF-MANUALE	14,00	1,00	15,00
CONF-ROVEMA	34,00	9,00	43,00
DECANTATORE-2	1,00		1,00
DECANTATORE-5	1,00		1,00
ESSICCATORE	10,00	4,00	14,00
ESTRATTORE	1,00		1,00
FILTRATRICE-1	4,00		4,00
IDROSOLUBILI	33,00		33,00
LIQCONFAUT1L-5L	49,00		49,00
LIQCONFMAN	4,00	4,00	8,00
MISC-FOGLIARI-1	11,00		11,00
MISC-FOGLIARI-2	12,00	1,00	13,00
PELLET	1,00		1,00
REATTORE-1	5,00		5,00
REATTORE-2	7,00		7,00
REATTORE-3	6,00		6,00
REATTORE-4	5,00		5,00
REATTORE-5	6,00		6,00
REATTORECHEL-01	2,00	1,00	3,00
REATTORECHEL-02	1,00		1,00
REATTORECHEL-03	2,00		2,00
SISTEMA-HA		6,00	6,00
Grand Total	215,00	26,00	241,00

Imponendo la visualizzazione dei primi 3 valori del campo "Codice risorsa" la pivot si rimpicciolisce.

I subtotali e i grand total si aggiornano sulla base degli elementi visualizzati. Nell'intestazione del campo filtrato col comando compare la scritta "[top 3]".
Abilitando "Mostra Altri" viene aggiunta una riga "Others" in cui vengono condensati tutti i valori rimanenti. I grand total in questo caso mostrano gli stessi valori visualizzati quando la pivot è completa.

Valenza simulata	Data ora inizio simulata		
Codice risorsa [top 3]	lug 2008	ago 2008	Grand Total
CONF-1KG	3,00		3,00
CONF-GRANULARI	3,00		3,00
CONF-MANUALE	14,00	1,00	15,00
Others	195,00	25,00	220,00
Grand Total	215,00	26,00	241,00

Modificando l'ordinamento del campo "Codice risorsa" cambiano anche gli elementi visualizzati. Qui "Codice risorsa" è ordinato in ordine alfabetico crescente:

Valenza simulata	Data ora inizio simulata		
Codice risorsa [top 3]	lug 2008	ago 2008	Grand Total
CONF-1KG	3,00		3,00
CONF-GRANULARI	3,00		3,00
CONF-MANUALE	14,00	1,00	15,00
Grand Total	20,00	1,00	21,00

Nella stessa pivot ora "Codice risorsa" è ordinato in ordine alfabetico decrescente:

Valenza simulata	Data ora inizio simulata		
Codice risorsa [top 3]	lug 2008	ago 2008	Grand Total
SISTEMA-HA		6,00	6,00
REATTORECHEL-03	2,00		2,00
REATTORECHEL-02	1,00		1,00
Grand Total	3,00	6,00	9,00

8.6.5 Toolbar standard pivot

I pulsanti visibili nella toolbar della pivot possono variare a seconda dell'applicazione o del tipo di report. I pulsanti descritti qui di seguito sono quelli della dotazione standard:

Opzioni

Attraverso il pulsante è possibile accedere al pannello di configurazione del report in cui è possibile definire viste, scenari, join, colonne custom, ecc...

Scenari

Nel menu a tendina è possibile selezionare quale scenario applicare al report.

Prefiltri

Nel menu a tendina è possibile selezionare quale prefiltro applicare ai dati.

Larghezza ottimale colonne

Il pulsante impone una larghezza ottimale alle colonne della griglia del gantt (il calcolo è approssimativo e non tiene conto di eventuali decorazioni sulla colonna).

Rinfresca report figli

Il pulsante serve a rinfrescare il contenuto degli eventuali report figli collegati al report pivot.

Esporta

Dal menu a tendina è possibile eseguire un export in vari formati del contenuto della pivot.

Stampa

Il comando di stampa apre l'anteprima di stampa del report pivot.

8.6.6 Customization Form

La "Customization Form" permette di aggiungere campi dalla pivot a runtime attraverso il drag and drop. E' invocabile dal menu contestuale della pivot (Show Field List).

8.6.7 Modifica della configurazione

La pivot utilizza diversi pannelli standard di configurazione. Di seguito verranno approfondite le configurazioni specifiche del report pivot.

8.6.7.1 Formule su aggregato (pivot)

La formula su aggregato è uno speciale campo data (numerico) che può essere customizzato dall'utente.

Questo campo può assumere solo valori numerici ed essere posizionato esclusivamente nella Data header area.

Il campo è definito attraverso una formula tra i valori delle celle di altri campi dati, quindi il risultato cambia al variare della struttura della pivot. Possono essere elaborati i valori sommarizzati (come Sum, Count, Min, ecc..) di altri campi data allo stesso livello gerarchico del campo Formula su aggregato.

Tempo lav simulato | Formula1 | Formula2 | Data ora inizio simulata | Data

Numero fase	Codice risorsa	14/07/2008 Tempo lav simulato	Formula1	Formula2	21/07/2008 Tempo lav simulato
5	CONF-1KG	8,00	16,00	4,00	
	MISC-FOGLIARI-1	16,27	32,53	16,27	
5 Sum		24,27			
5 Average		8,09			
5 Custom			48,53	8,09	

Configurazione Pivot

Viste
Scenari
Decorazioni
Joins
Colonne custom
Formule su aggregato

Formule su aggregato

Nome	Formula
Formula1	CellSum('TEMPO_LAV_SIMULATO') * 2
Formula2	CellAvg('TEMPO_LAV_SIMULATO')

Le formule su aggregato possono anche contenere riferimenti ai subtotali (locali al livello corrente) di un campo data o ai grand total (assoluti) di un campo data. Questi sono utili per poter esprimere dei valori come percentuale sul totale come in questo esempio.

Le formule su aggregato possono anche contenere riferimenti ai valori della cella precedente la cella attuale.
Per far questo bisogna aggiungere '-1' come secondo parametro delle funzioni che estraggono i valori delle celle, ad esempio CellSum ('campo'; '-1').
L'esempio Calcolo del montante utilizza queste espressioni per calcolare il montante di un certo valore.

8.6.7.1.1 Esempio: Calcolo del montante

Vediamo come utilizzare una formula su aggregato per calcolare il montante di un valore aggregato (estratto dal DB Nicim).
Costruiamo una pivot che sommarizza a settimane i PEZZI_BUONI registrati nella tabella PROD per una certa risorsa:

Pivot da query

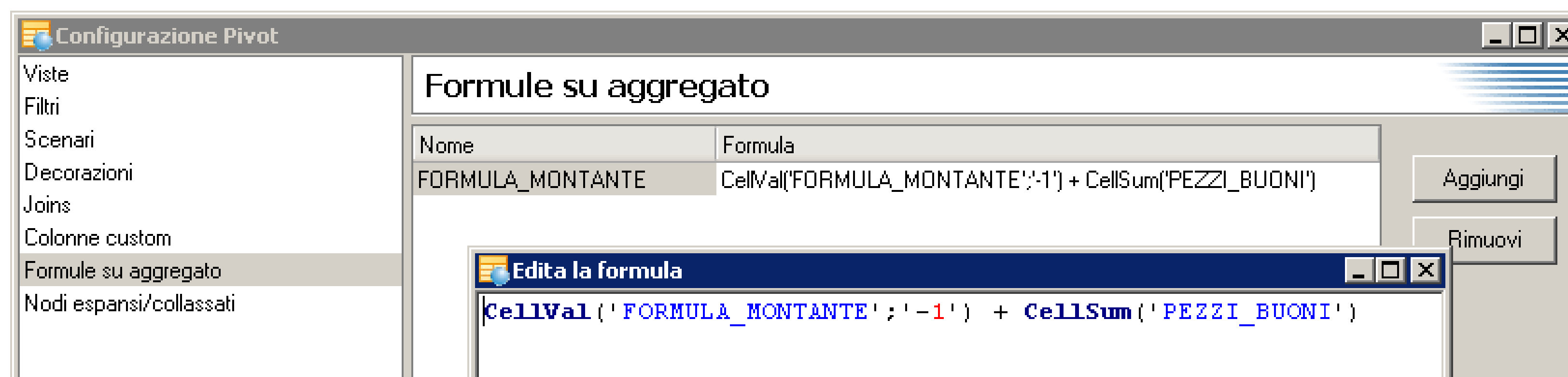
Scenario | (Nessuno)

Drop Filter Fields Here

Pezzi buoni | Data

Risorsa prevista	27/03/2006	03/04/2006	10/04/2006	Grand Total
ESSICCATORE	5000,00	28500,00	18135,00	51635,00

Aggiungiamo una formula su aggregato di nome FORMULA_MONTANTE costruita con questa espressione:
CellVal('FORMULA_MONTANTE';'-1') + CellSum('PEZZI_BUONI')



Aggiungiamo il nuovo campo FORMULA_MONTANTE alla pivot:

Pivot da query

Scenario (Nessuno)

Drop Filter Fields Here

Pezzi b | FORMULA_MON | Data | Data

Risorsa prevista	27/03/2006		03/04/2006		10/04/2006		Grand Total	
	Pezzi buoni	FORMULA_MONTA	Pezzi buoni	FORMULA_MONTA	Pezzi buoni	FORMULA_MONTA	Pezzi buoni	FORMULA_MONTA
ESSICCATORE	5000,00	5000,00	28500,00	33500,00	18135,00	51635,00	51635,00	51635,00

Normalmente il CellValue della cella precedente viene calcolato nella direzione in cui si trova il Data Field Anchor (orizzontale se il Data Field Anchor è nella Column Area, verticale se il Data Field Anchor è nella Row Area).

E' possibile però definire in modo esplicito la direzione verso cui estrarre la celle precedente utilizzando 2 formati diversi del comando

CellValue. Rifacendosi all'esempio qui sopra:

CellVal('FORMULA_MONTANTE';'-X') ritorna il valore della cella FORMULA_MONTANTE che precede orizzontalmene

CellVal('FORMULA_MONTANTE';'-Y') ritorna il valore della cella FORMULA_MONTANTE che precede verticalmente

Esempio di montante orizzontale:

Pezzi buoni | MONTANTE_ORIZ | Codice ordine | Data | Data

Risorsa prevista	200600028772					
	27/03/2006		03/04/2006		10/04/2006	
	Pezzi buoni	MONTANTE_ORIZ	Pezzi buoni	MONTANTE_ORIZ	Pezzi buoni	MONTANTE_ORIZ
ESSICCATORE	3500,00	3500,00	14000,00	17500,00	3435,00	20935,00
REATTORECHEL-03	3500,00	3500,00	14000,00	17500,00	3435,00	20935,00
SISTEMA-HA	3500,00	3500,00	14000,00	17500,00	3435,00	20935,00

Formula per MONTANTE_ORIZ : CellVal('MONTANTE_ORIZ';'-x') + CellSum('PEZZI_BUONI')

Esempio di montante verticale:

Pezzi buoi | MONTANTE_VEF | Codice ordine | Data | Data

Risorsa prevista	200600028772					
	27/03/2006		03/04/2006		10/04/2006	
	Pezzi buoni	MONTANTE_VERT	Pezzi buoni	MONTANTE_VERT	Pezzi buoni	MONTANTE_VERT
ESSICCATORE	3500,00	3500,00	14000,00	14000,00	3435,00	3435,00
REATTORECHEL-03	3500,00	7000,00	14000,00	28000,00	3435,00	6870,00
SISTEMA-HA	3500,00	10500,00	14000,00	42000,00	3435,00	10305,00

Formula per MONTANTE_VERT : CellVal('MONTANTE_VERT';'-y') + CellSum('PEZZI_BUONI')

8.6.7.1.2 Esempio: Calcolo di percentuale su totale

Vediamo come utilizzare una formula su aggregato per calcolare la percentuale sul totale di un valore aggregato (estratto dal DB Nicim). Costruiamo una pivot che sommarizza a settimane i PEZZI_BUONI registrati nella tabella PROD per una certa risorsa:

Pivot da query

Scenario | (Nessuno)

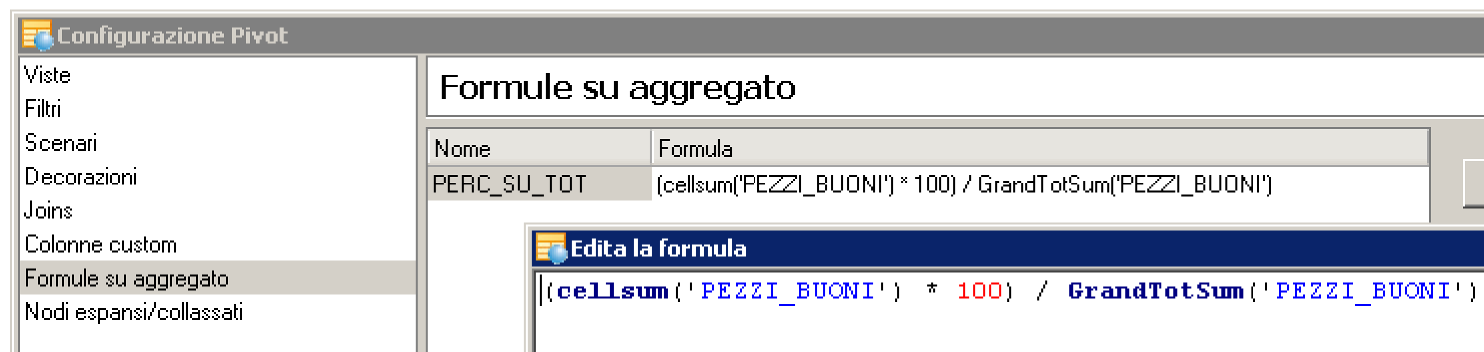
Drop Filter Fields Here

Pezzi buoni | Data

Risorsa prevista	27/03/2006	03/04/2006	10/04/2006	Grand Total
ESSICCATORE	5000,00	28500,00	18135,00	51635,00

Aggiungiamo una formula su aggregato di nome PERC_SU_TOT costruita con questa espressione:

(CellSum('PEZZI_BUONI') * 100) / GrandTotSum('PEZZI_BUONI')



Aggiungiamo il nuovo campo PERC_SU_TOT alla pivot:

Pivot da query

Scenario | (Nessuno)

Drop Filter Fields Here

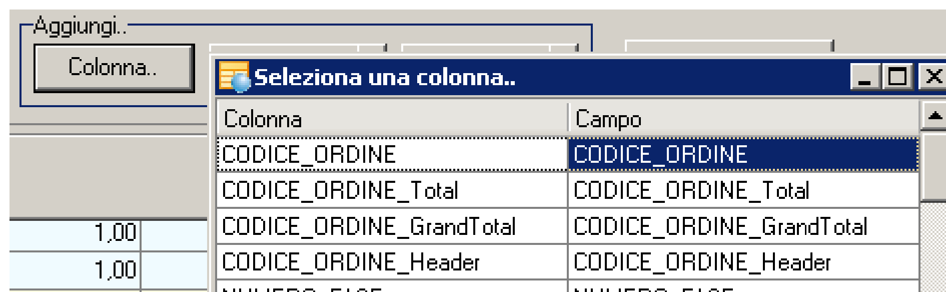
Pezzi buoni | PERC_SU_TOT | Data | Data

Risorsa prevista	27/03/2006		03/04/2006		10/04/2006		Grand Total	
	Pezzi buoni	PERC_SU_TOT	Pezzi buoni	PERC_SU_TOT	Pezzi buoni	PERC_SU_TOT	Pezzi buoni	PERC_SU_TOT
ESSICCATORE	5000,00	9,68	28500,00	55,20	18135,00	35,12	51635,00	100,00

8.6.7.2 Decorazioni di celle (pivot)

Nella pivot è possibile creare delle decorazioni non solo sulle celle dei valori aggregati della Data area ma anche sui totali, sui subtotali e sulle intestazioni di celle e colonne.

Le decorazioni vengono create tutte nello stesso modo descritto precedentemente dopo averne selezionato il target cliccando sul tasto "Colonna..":



Nelle formule delle decorazioni di celle, totali e subtotali è possibile utilizzare il valore della cella corrente (CellValue) e i valori delle intestazioni della riga e della colonna relativi alla cella corrente (ColHeader0-9 e RowHeader0-9).
Nelle formule delle decorazioni delle intestazioni è possibile utilizzare il solo valore della cella corrente (CellValue) ed ottenere risultati simili a questo:

8.6.7.3 Prefiltri (pivot)

I prefiltri sono dei filtri che vengono eseguiti sulla tabella che alimenta la pivot.

I prefiltri quindi non filtrano i dati attualmente visualizzati ma filtrano i dati nella loro forma tabellare originaria prima dell'aggregazione compiuta dalla pivot.

Il prefiltro attualmente utilizzato è visualizzato ai piedi della pivot.

Quantita teorica | Data disp | Codice fase

Causale mov	Codice ordine	Codice parte	21/04/2006 30	21/04/2006 999	21/04/2006 Total
C	200600038935	010131468	2331,00		2331,00
	200600038935 Total		2331,00		2331,00
C Count			1	1	2
P	200600038935	0101114681801		1850,00	1850,00
	200600038935 Total			1850,00	1850,00

(Quantita teorica > 500) Prefilter...

I prefiltri possono essere inseriti durante la configurazione dei report ed utilizzati all'interno degli scenari.

Possono essere anche creati o modificati "al volo" attraverso il pulsante "Prefilter.." che compare a fianco del prefiltro attivo o cliccando sulla voce "Configura prefiltro.." di uno dei menu contestuali dell'area header della pivot o cliccando sulla voce "Custom prefilter.." di un altro dei menu contestuali dell'area header. In questo caso le modifiche non sono persistenti e non vengono salvate su disco.

8.6.7.4 Colonne custom (pivot)

Le colonne custom sono campi calcolati che vengono aggiunti alla tabella che alimenta la pivot.

Le formule sono risolte a **livello di tabella di alimentazione** e quindi i campi risultanti nella pivot sono equiparati ai campi originali della tabella di alimentazione.

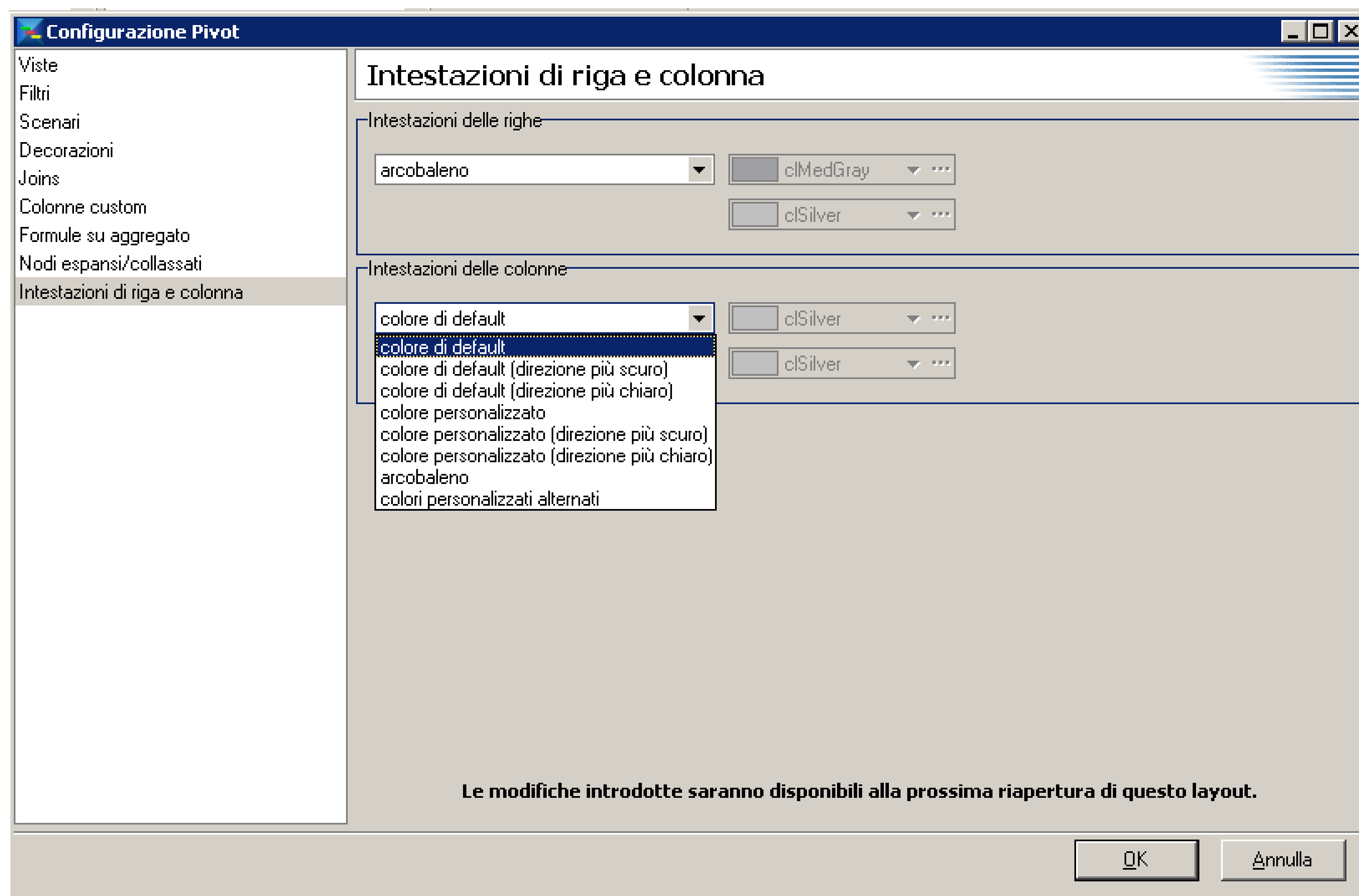
8.6.7.5 Intestazioni di riga e colonna (pivot)

E' possibile evidenziare le instestazioni di riga e/o colonna in base al loro livello nella gerarchia di aggregazione.

Le celle delle intestazioni possono essere dipinte con un colore di sfondo che sfuma verso il chiaro o lo scuro in base al livello di aggregazione a cui appartiene la cella. Possono anche essere colorate con colori alternati.

Nell'apposito pannello di configurazione è possibile impostare due configurazioni differenti per le intestazioni di riga e per quelle di

colonna:


Configurazione Pivot
Viste
Filtri
Scenari
Decorazioni
Joins
Colonne custom
Formule su aggregato
Nodi espansi/collassati
Intestazioni di riga e colonna
Intestazioni di riga e colonna
Intestazioni delle righe
arcobaleno
clMedGray
clSilver
Intestazioni delle colonne
colore di default
clSilver
clSilver
colore di default
colore di default (direzione più scuro)
colore di default (direzione più chiaro)
colore personalizzato
colore personalizzato (direzione più scuro)
colore personalizzato (direzione più chiaro)
arcobaleno
colori personalizzati alternati
Le modifiche introdotte saranno disponibili alla prossima riapertura di questo layout.
OK
Annulla

L'effetto arcobaleno sulle intestazioni di riga:

T Lav Ore | Partizione

Risorsa	Fase	Sottofase	T Avv Ore	N Pezzi	Ordine	CustomCol1	17/06/2010	18/
CDL10	10	1	,00	12000	200600015942	1	4,8	
					200600015942 Total		4,8	
				12000 Total			4,8	
			,00				4,8	
		1 Total					4,8	
	10 Total						4,8	
CDL10 Total							4,8	
					200600041168	0		

8.6.7.6 Modifica della configurazione in modalità utente (pivot)

Quando l'applicazione non si trova in modalità configuratore, le opzioni di personalizzazione della pivot si riducono e l'utente può accedere ad un pannello di configurazione semplificato rispetto allo standard.

In questo pannello l'utente può modificare:

- font dei titoli
- font delle celle della pivot

Cliccando sul pulsante Reset le modifiche dell'utente vengono eliminate e viene ristabilità la configurazione predefinita.

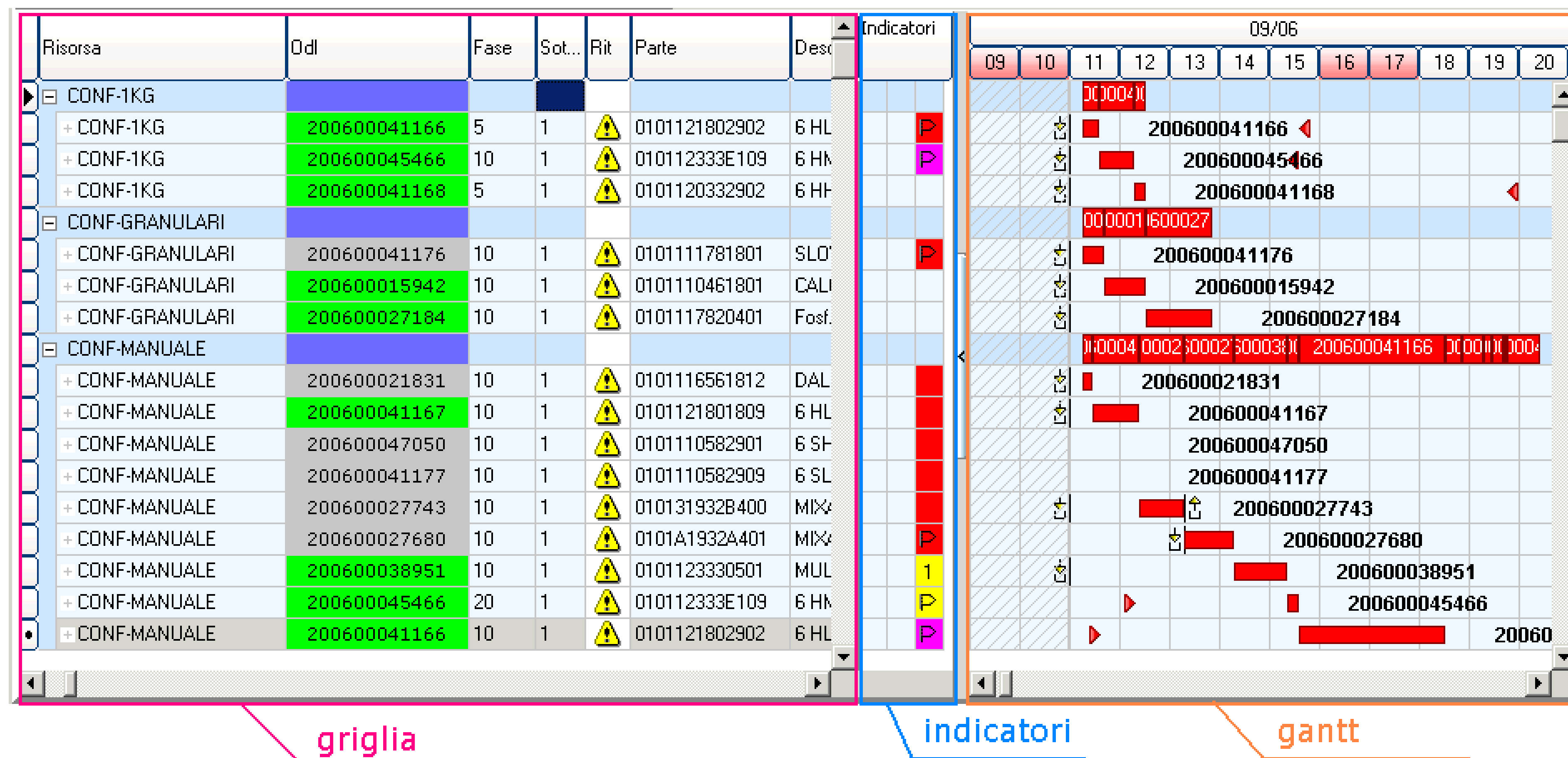
8.7 Report gantt

Il gantt è un particolare tipo di grafico a barre che illustra l'andamento temporale di attività ed eventi. Le singole attività sono rappresentate come barre con una data di inizio ed una data di fine. Le attività possono essere organizzate con legami di dipendenza.

Gantt chart su wikipedia

8.7.1 Struttura gantt

Il report gantt è organizzato in 3 aree:



A sinistra l'area griglia contenente le informazioni presentate in formato tabellare organizzate secondo la gerarchia attuale.

La griglia è caratterizzata da una serie di colonne di cui solitamente la prima ha una struttura ad albero e può essere utilizzata per la navigazione livello per livello all'interno dei dati del gantt.

Quando un nodo della gerarchia del gantt viene imploso (cliccando sull'apposito pulsante), tutti i nodi dei livelli sottostanti vengono nascosti così come le rispettive barre sull'area gantt.
I livelli pari e i livelli dispari dell'area griglia posseggono attributi di colore di sfondo separati. In questo modo l'utente può definire combinazioni di colori che aumentano la leggibilità dei livelli.
Lo stesso schema di colori viene riproposto anche nell'area indicatori e nell'area gantt.

Al centro l'area indicatori che contiene un elenco di indicatori grafici proprietari dell'applicazione. Gli indicatori sono built-in nell'applicazione e sono utilizzati per fornire delle informazioni che devono essere visibili a colpo d'occhio. Gli indicatori, se non utilizzati nell'analisi del dato, possono comunque essere nascosti dall'utente.
L'intera area degli indicatori può essere minimizzata cliccando sul pulsante che si trova al centro tra l'area degli indicatori e l'area del gantt. Quando è minimizzata quest'area può essere ripristinata cliccando sulla piccola intestazione che rimane sempre visibile tra l'area griglia e l'area gantt.
Quando il report viene stampato, l'area indicatori non viene stampata.

A destra l'area gantt vera e propria dove vengono visualizzate le barre temporali allineate ad una barra dei tempi visibile in alto.

8.7.1.1 Barre gantt

Esistono 2 tipi di barre del gantt:

- le **barre vere e proprie** caratterizzate da una data di inizio e una data di fine, rappresentate sul gantt come un rettangolo
- le **milestone** che fanno riferimento ad un'unica data e sono rappresentate da un'icona

In generale ogni applicazione presenta un set di barre predefinite a cui l'utente può aggiungere a piacere una o più barre custom.

Ad ogni barra è possibile associare un campo della tabella di alimentazione del report che faccia funzione di **etichetta**. Il valore del campo viene pubblicato a fianco della barra nell'area gantt (è possibile customizzarne l'aspetto grafico e la posizione relativa).

Una barra che appartiene ad un certo livello può essere resa visibili anche al livello superiore nella cosiddetta **barra del raggruppato**. La barra del raggruppato è una combinazione di tutte le barre appartenenti al livello inferiore ed può essere visualizzata solo se il livello attuale non possiede barre proprie. Le milestone sono escluse dalla barra del raggruppato.
La barra del raggruppato può essere rappresentata con un'unica barra che inizia quando inizia la prima (temporalmente parlando) barra dei nodi figli e finisce quando finisce l'ultima barra dei nodi figli, oppure può essere rappresentata da un duplicato di tutte le barre dei nodi figli riproposte sul livello superiore.

8.7.2 Menu contestuale standard gantt

Il report gantt standard propone un menu contestuale invocabile sull'area griglia da cui richiamare i seguenti comandi:

- **Copia negli appunti**: copia il contenuto della cella corrente della griglia del gantt negli appunti di Windows.
- **Trova**: apre la finestra di ricerca standard che permette di cercare le occorrenze di un valore all'interno della colonna corrente della griglia del gantt.

8.7.3 Toolbar standard gantt

I pulsanti visibili nella toolbar del gantt possono variare a seconda dell'applicazione o del tipo di report. I pulsanti descritti qui di seguito

sono quelli della dotazione standard:

Opzioni

Attraverso il pulsante è possibile accedere al pannello di configurazione del report gantt in cui è possibile definire viste, scenari, join, colonne custom, ecc...

Scenari

Nel menu a tendina è possibile selezionare quale scenario applicare al report.

Espandi/Collassa

Il pulsante espande tutti i nodi del gantt, mentre il pulsante li collassa.

Larghezza ottimale colonne

Il pulsante impone una larghezza ottimale alle colonne della griglia del gantt (il calcolo è approssimativo e non tiene conto di eventuali decorazioni sulla colonna).

Trova avanzato

E' possibile definire delle ricerche complesse sui dati visualizzati nella griglia del gantt. Attraverso il pannello del trova avanzato l'utente può definire delle condizioni di ricerca su più campi e con operatori logici. I nodi del gantt che soddisfano la condizione espressa vengono contrassegnati da un rettangolino giallo disposto in un'area particolare della griglia del gantt che compare a sinistra ogni qual volta una ricerca viene attivata.

L'utente può navigare nell'elenco dei nodi trovati cliccando sui rettangoli visualizzati. In alternativa può scorrere i risultati attraverso i due pulsanti della toolbar: e .

Stampa

Il comando di stampa apre l'anteprima di stampa del gantt.

8.7.4 Modifica della configurazione

Il gantt utilizza diversi pannelli standard di configurazione. Di seguito verranno approfondite le configurazioni specifiche del report gantt.

8.7.4.1 Configurazione delle barre del gantt

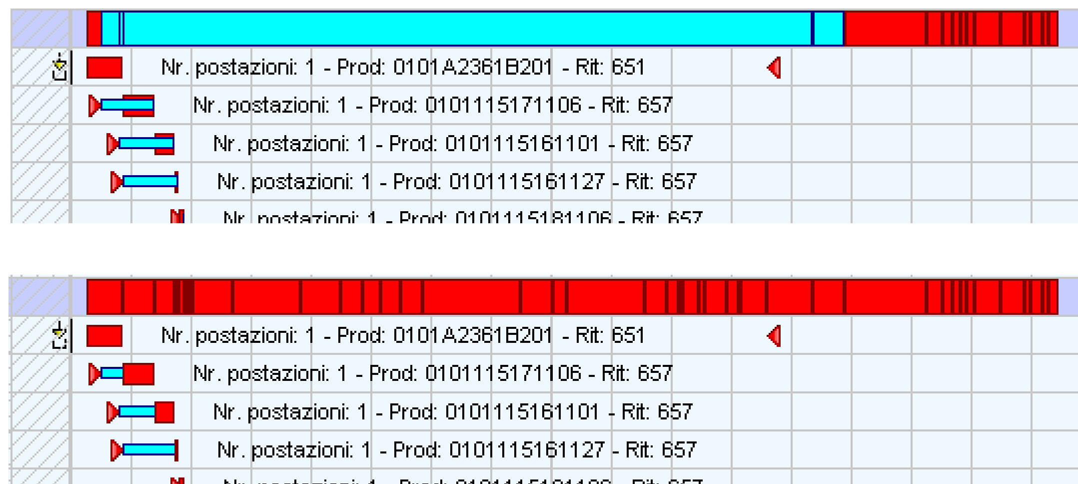
E' possibile definire un set di impostazioni per le barre del gantt.

Per ogni set di barre l'utente può modificare:

- l'altezza delle righe del gantt (espressa in pixel)
- la visibilità della barra del raggruppato ed eventualmente il tipo (barra unica o replica delle barre dei nodi figli). In caso di replica, è possibile visualizzare anche l'etichetta delle barre figlie che comparirà all'interno della barra replicata.

Di ogni barra definita l'utente può modificare alcune proprietà:

- visibilità
- l'ordine di disegno. Le barre vengono dipinte nell'ordine espresso da questo valore. E' importante in caso di barre che si sovrappongono (sui nodi terminali o sulla barra del raggruppato) per garantire che sia visibile l'informazione più importante per l'analisi (ad esempio che sia sempre visibile la barra che indica un ritardo). Stesso gantt con ordini di disegno differenti (la barra celeste è dipinta dopo la barra rossa nel primo, nel secondo gantt l'ordine è invertito):



- l'etichetta. E' possibile definire quale colonna della tabella in alimentazione deve fornire i valori che vengono utilizzati come etichetta delle barre. E' possibile scegliere il font e la posizione relativa dell'etichetta.
- le decorazioni della barra (non milestone). E' possibile esprimere una o più condizioni che, quando validate, impongono alla barra un colore di sfondo e di bordo alternativo al colore di default. La condizione è espressa attraverso una formula secondo un meccanismo condiviso con le decorazioni delle celle.
- è possibile far colorare le barre (non milestone) sulla base dei valori assunti da un campo della tabella di alimentazione. Questo meccanismo si chiama "traccia". Le barre sulle righe che hanno il medesimo valori di un certo campo scelto dall'utente vengono colorate tutte con lo stesso colore. L'utente può scegliere quale colonna andrà "tracciata". I colori vengono generati automaticamente dall'applicazione con un meccanismo randomico, l'utente può comunque definire un livello di luminosità e di saturazione che verranno utilizzati durante la generazione dei colori.

8.7.4.2 Configurazione degli indicatori del gantt

E' possibile nascondere a piacere gli indicatori del gantt built-in nell'applicazione.

8.7.4.3 Configurazione delle barre custom

L'utente può avere la necessità di definire delle barre nell'area gantt oltre a quelle fornite built-in dall'applicativo.

Dopo aver dato un nome univoco alla barra, l'utente deve decidere se la barra custom è una barra normale o una milestone.

Se è una barra normale è necessario definire:

- quale campo della tabella di alimentazione deve essere usato come data inizio e quale come data fine;
- il colore di sfondo e il colore del bordo;
- se la barra deve essere visibile sull'eventuale barra del raggruppato;
- l'altezza della barra custom e la posizione in pixel rispetto al centro della riga.

Se è una milestone è necessario definire:

- quale campo della tabella di alimentazione deve essere usato come data;
- quale icona deve essere usata come milestone (può essere scelta da un elenco o importata da file).

Le barre custom appena definite sono disponibili alla successiva riapertura del report.

8.7.4.4 Modifica della configurazione in modalità utente

In modalità utente, la configurazione modificabile dell'utente è limitata ad un set di impostazioni ristretto:

- visibilità delle colonne nella griglia del gantt
- font dei nodi dispari e dei nodi pari nella griglia del gantt
- altezza delle righe della griglia del gantt

Cliccando sul pulsante Reset le modifiche dell'utente vengono eliminate e viene ristabilità la configurazione predefinita.

8.8 Report albero

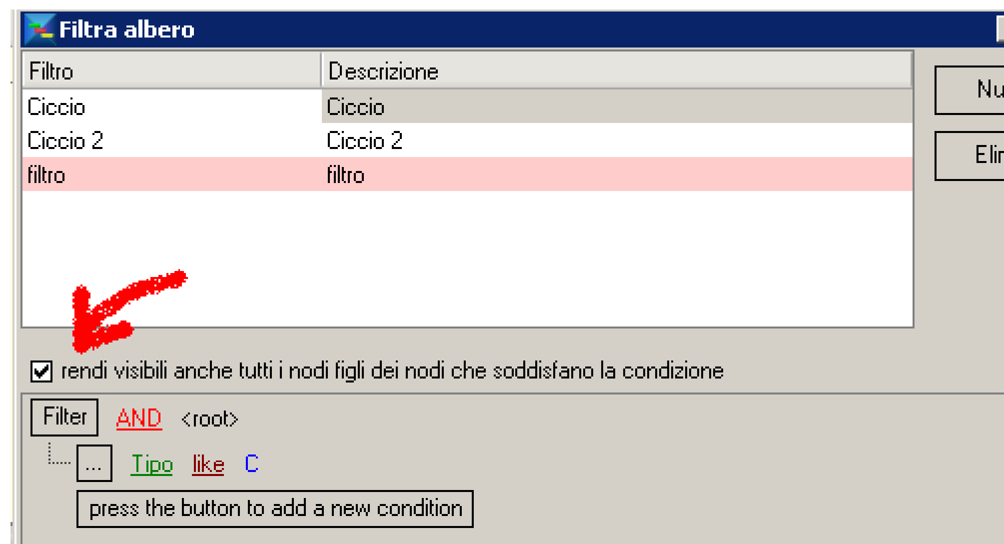
L'albero è un report utilizzabile per visualizzare una gerarchia di dati.

8.8.1 Filtri del report albero

Il report ad albero è utilizzato per visualizzare una gerarchia.

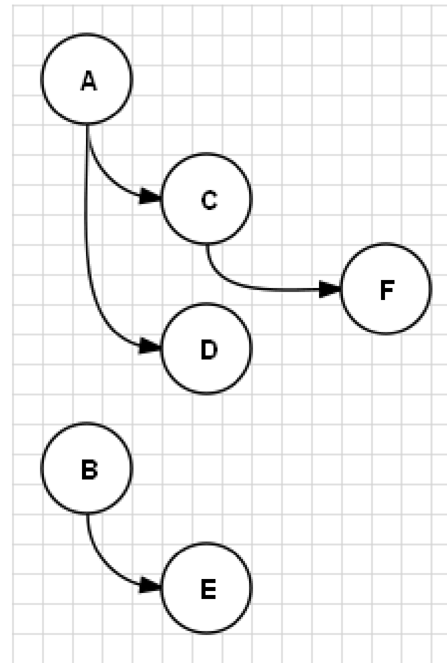
Per questo motivo i filtri inseriti in un report albero permettono di visualizzare un sotto-albero dello stesso secondo questa procedura:

1. vengono nascosti tutti i nodi dell'albero
2. vengono resi visibili tutti i nodi dell'albero che soddisfano la condizione del filtro
3. vengono resi visibili i nodi dei percorsi minimi per raggiungere i nodi precedentemente marcati come visibili al punto 1 a partire dai nodi radice dell'albero
4. se nel pannello di modifica del filtro è stato scelta l'opzione "rendi visibili anche tutti i nodi figli dei nodi che soddisfano la condizione" vengono resi visibili tutti i nodi figli (di ogni livello) dei nodi visibili calcolati al punto 1

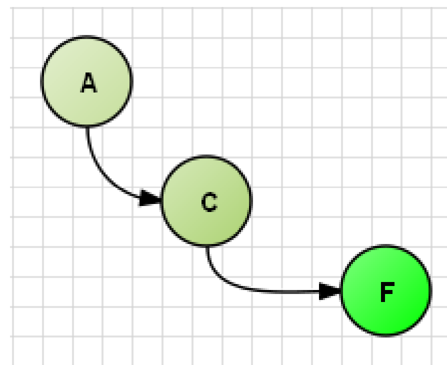


Vediamo qualche esempio.

Partiamo da un albero così costruito:



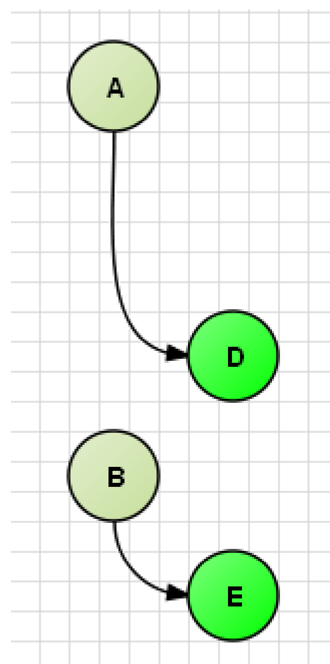
Se il filtro è **NODO=F**, il risultato è:



Il nodo F è visibile perchè soddisfa la condizione, i nodi A e C sono visibili perchè rappresentano il percorso minimo per arrivare al nodo F

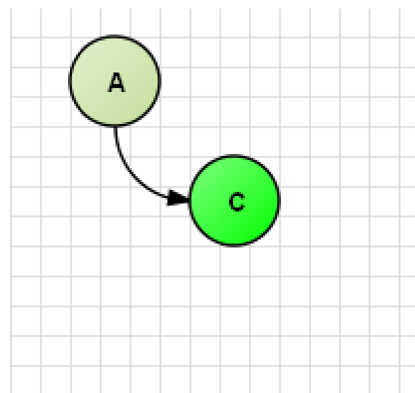
partendo dalle radici.

Se il filtro è **(NODO = D) OR (NODO=E)**, il risultato è:



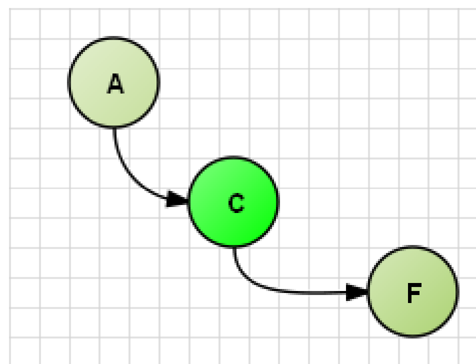
I nodi D ed E sono visibili perchè soddisfano la condizione, il nodo A è visibile perchè è il percorso minimo per arrivare al nodo D ed il nodo B è visibile perchè è il percorso minimo per arrivare al nodo E.

Se il filtro è **NODO=C**, il risultato è:



Il nodo C è visibile perchè soddisfa la condizione, il nodo A è visibile perchè appartiene al percorso minimo per raggiungere il nodo C partendo dalle radici.

Se il filtro è **NODO=C** ed è marcato il check "rendi visibili anche tutti i nodi figli dei nodi che soddisfano la condizione" il risultato è:



Il nodo C è visibile perchè soddisfa la condizione, il nodo A è visibile perchè appartiene al percorso minimo per raggiungere il nodo C partendo dalle radici, il nodo F è visibile perchè è figlio di C ed il check "rendi visibili anche tutti i nodi figli dei nodi che soddisfano la condizione" è stato marcato.

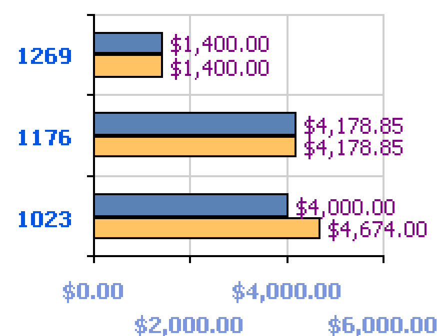
8.9 Report grafico

Questo report è in grado di disegnare un grafico (istogramma, diagramma a torta, ecc..) a partire da un set di dati.

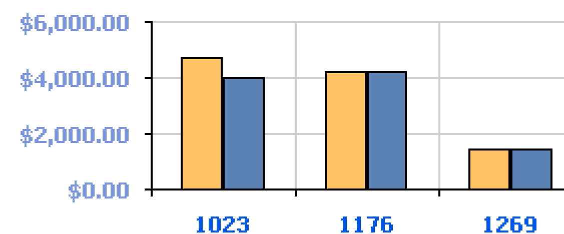
I tipi di grafico disponibili sono:

- diagramma a barre
- diagramma a colonne
- diagramma a linee
- diagramma ad area
- diagramma a torta

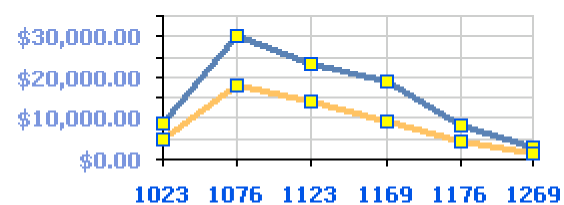
Bar Diagram



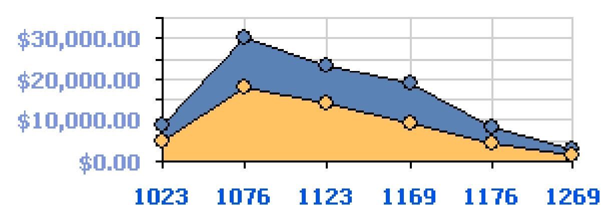
Column Diagram



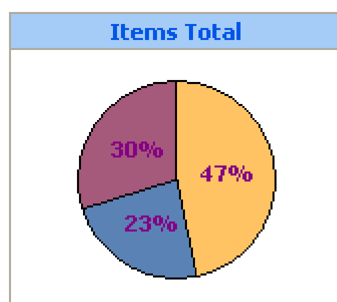
Line Diagram



Area Diagram

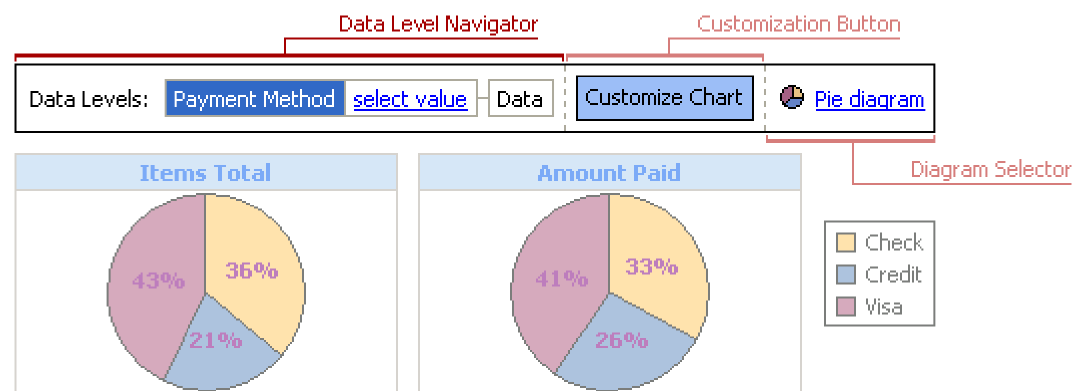


Pie Diagram



Nel report è visualizzato un chart toolbox che è un pannello da cui è possibile accedere ad alcune funzioni:

- la customizzazione delle proprietà grafiche e dell'aspetto del report (diagram selector e chart customization button)
- il meccanismo di data-mining dinamico (il data level navigator)



8.9.1 Struttura grafico

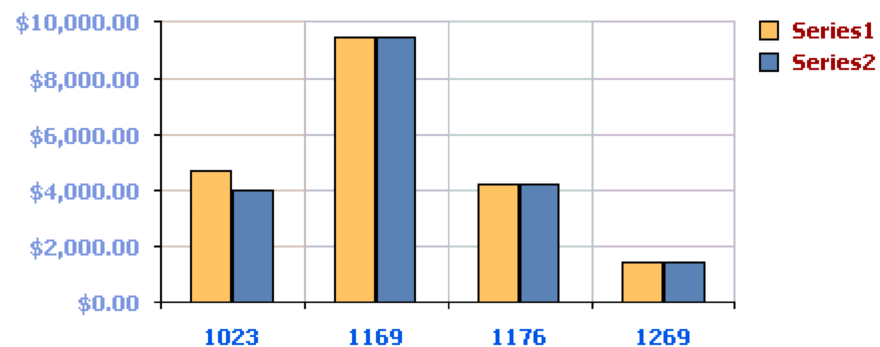
Le serie, i data group e le categorie sono gli elementi base dei grafici.

La **serie** rappresenta un set di valori che vengono disegnati in un grafico.

Ogni serie ha un colore o un pattern unico ed è elencata nella legenda del grafico.

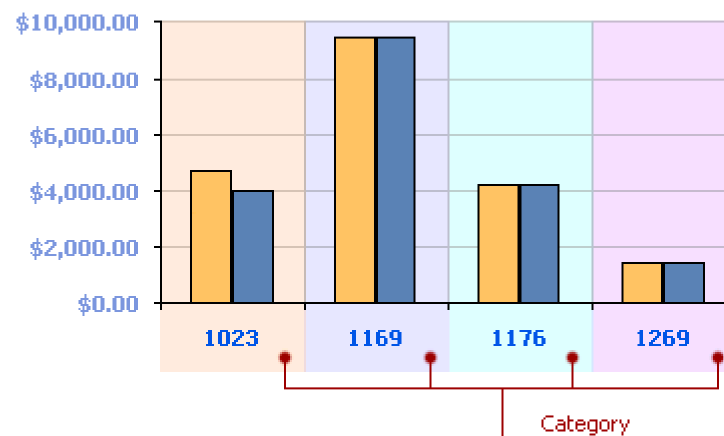
Normalmente i valori delle serie visualizzate in un grafico sono raggruppati in categorie.

In un grafico è possibile disegnare una o più serie per realizzare diagrammi semplici o multipli (raggruppati).

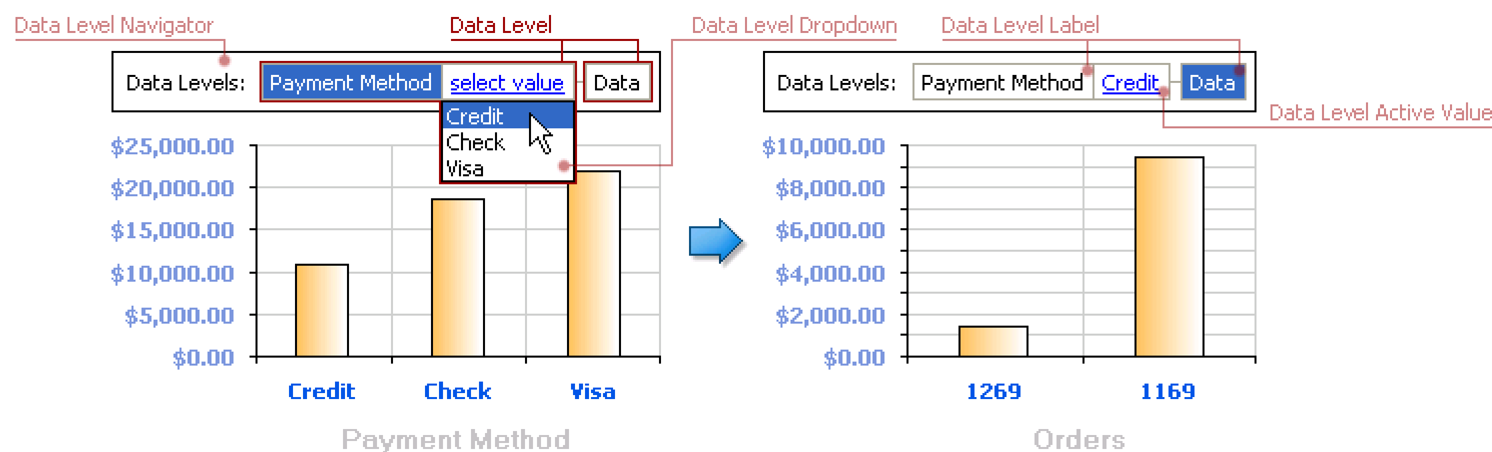


Le **categorie** servono a classificare i valori delle serie a scopo comparativo.

Le categorie sono equivalenti ai raggruppamenti realizzabili sui report griglia. Di default una categoria è costruita dai valori univoci di una delle colonne dei dati in input.

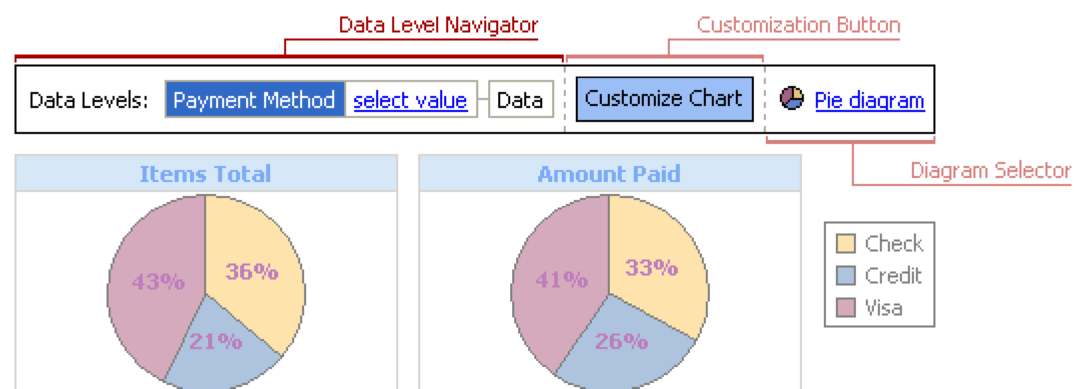


I **data group** sono uno strumento per configurare e poi effettuare il drill-down dei dati visualizzati nel grafico. Si definiscono a partire dal chart customization button possono poi essere navigati attraverso il data level navigator.



8.9.1.1 Diagram selector

Il diagram selector è uno dei tool di configurazione presenti nella chart toolbox.

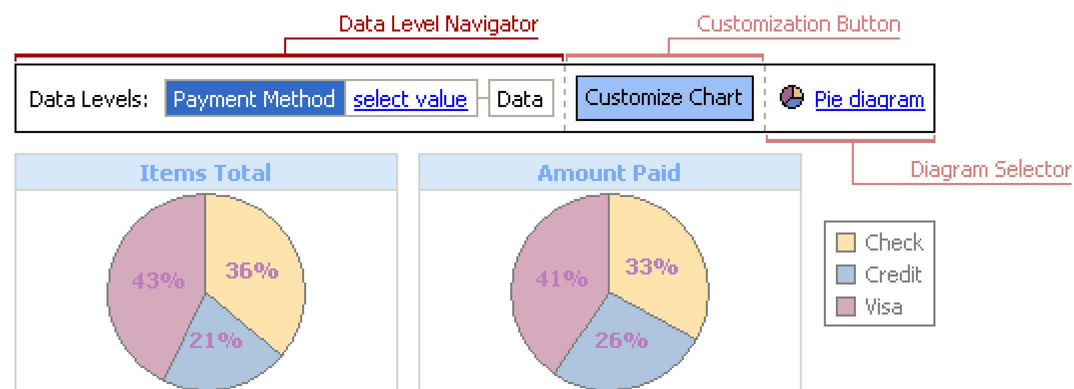


Il diagram selector serve a cambiare al volo il tipo di diagramma utilizzato e mostra quello corrente.

Cliccando sul diagram selector comparirà un elenco dei diaframmi disponibili.

8.9.1.2 Chart customization button

Il char customization button è accessibile dalla chart toolbox.



L'utente può cliccare su questo pulsante per visualizzare la customization form, da cui è possibile impostare alcune proprietà del report grafico.

La customization form ha 3 linguette:

Series

Questa pagina permette di definire quale serie debba essere visualizzata nel grafico, quale debba essere l'ordine di disegno delle serie visualizzate e come debbano essere ordinati i valori delle serie.

Data Groups

Questa pagina permette all'utente di mostrare o nascondere i data group con il drag'n'drop.

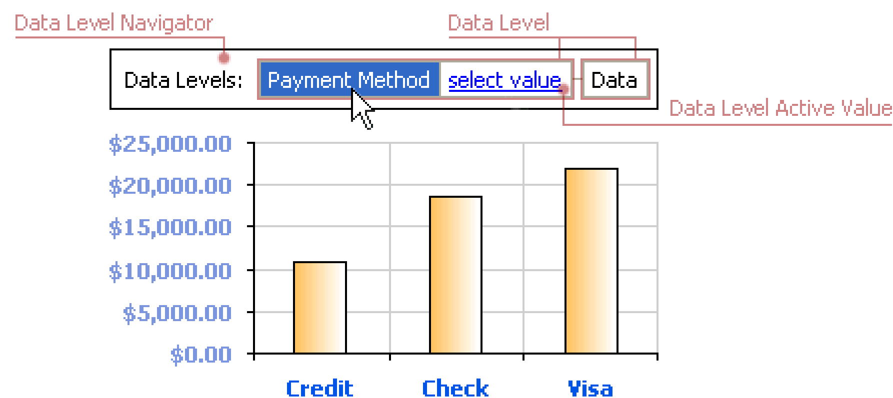
Options

Questa pagina permette di definire l'aspetto di alcuni elementi del grafico come la sua legenda, il titolo e la toolbox.

8.9.1.3 Data level navigator

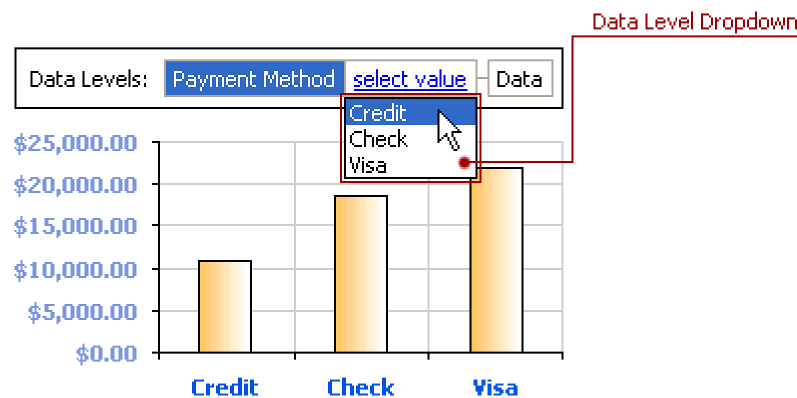
I **data group** servono ad effettuare un drill down attraverso i dati visualizzati sul grafico controllando il livello di dettaglio dei dati disegnati.

Un **data level** è la rappresentazione grafica di un data group ed è visibile nel data level navigator



Graficamente un data level consiste di questi elementi:

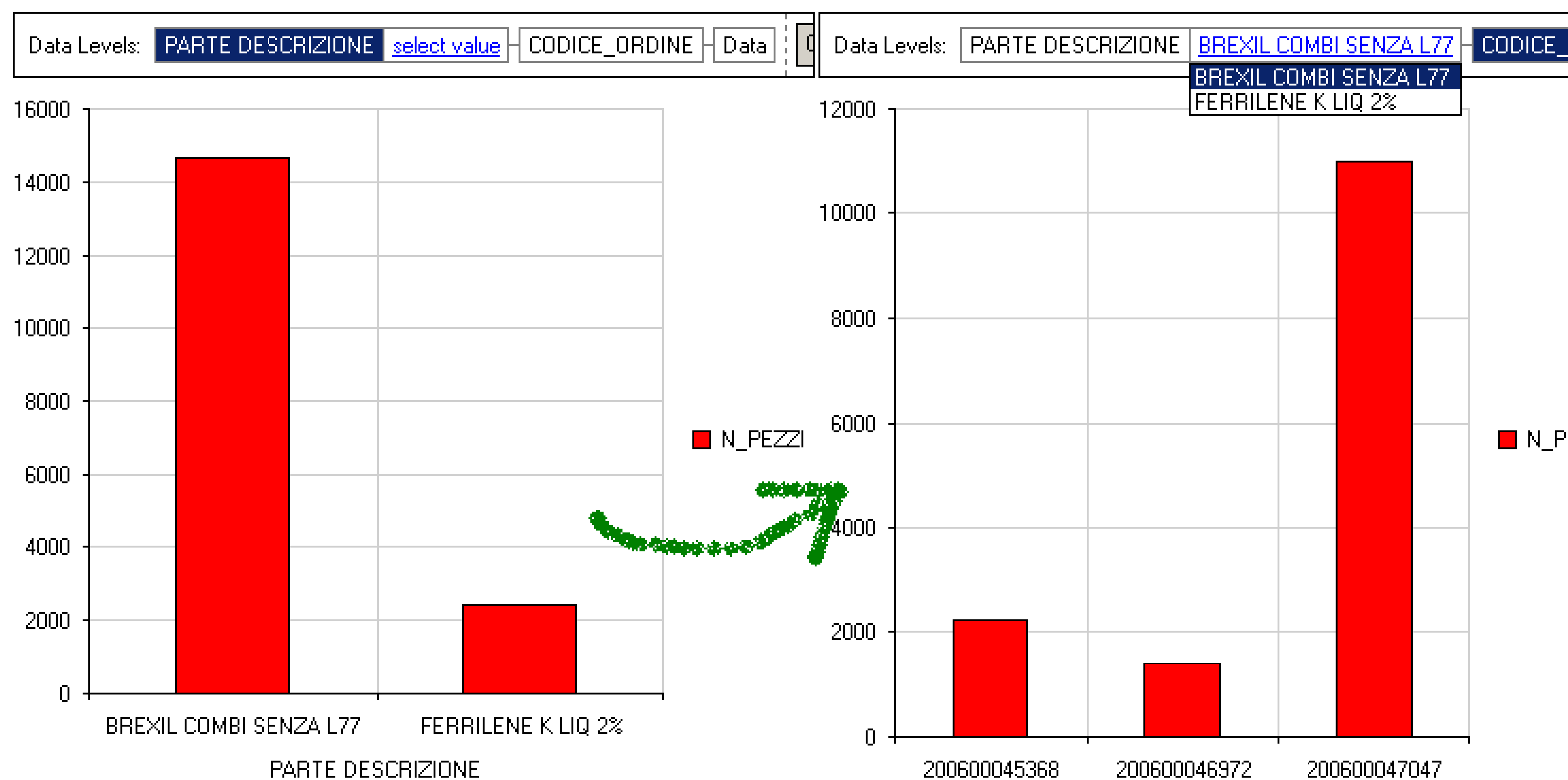
- una etichetta: cliccando su di essa si attiva il data level che così apparirà evidenziato.
- un valore per il data level: cliccando su di esso verrà visualizzato un menu a tendina contenente i valori possibili per il data level.

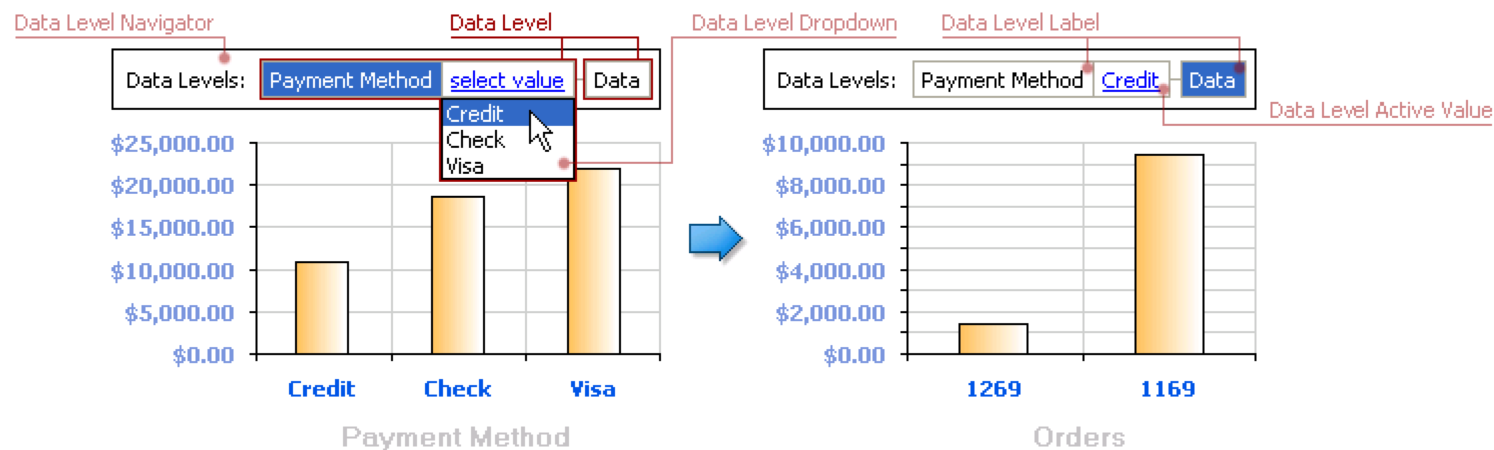


Il data level navigator è uno dei tool presenti nel chart toolbox che mostra la gerarchia dei data level e provvede a fornire un meccanismo interattivo per il data mining.

Attraverso di esso l'utente può:

- effettuare un drill-down aumentando il dettaglio dei dati selezionando un valore per il data level nel menu a tendina
- cambiare il livello di dettaglio dei dati visualizzati assegnando un valore attivo ai data level.
- definire una gerarchia di data levels riordinandoli, nascondendoli o aggiungendone di nuovi tutti attraverso il drag'n'drop da e verso la customization form.


Data Levels: PARTE DESCRIZIONE select value CODICE_ORDINE Data
16000
14000
12000
10000
8000
6000
4000
2000
0
BREXIL COMBI SENZA L77
FERRILENE K LIQ 2%
PARTE DESCRIZIONE
N_PEZZI
Data Levels: PARTE DESCRIZIONE BREXIL COMBI SENZA L77 CODICE_
BREXIL COMBI SENZA L77
FERRILENE K LIQ 2%
12000
10000
8000
6000
4000
2000
0
200600045368
200600046972
200600047047
N_P



8.9.2 Toolbar standard grafico

Opzioni

Attraverso il pulsante è possibile accedere al pannello di configurazione del report in cui è possibile definire scenari, join e colonne custom.

Scenari

Nel menu a tendina è possibile selezionare quale scenario applicare al report. Cliccando sulla voce "Modifica" si attiva la customization form del grafico.

Stampa

Il comando di stampa apre l'anteprima di stampa del report grafico.

8.10 Stampa report

Alcuni report (griglia, pivot e albero) condividono il sistema di gestione della stampa. Il report gantt ha un suo sistema di di stampa.

8.10.1 Stampa griglia, pivot, albero, grafico

Le impostazioni delle opzioni di stampa delle griglie e degli alberi, si definiscono attraverso la toolbar che si presenta all' apertura della finestra "Print Preview".

Il design report ( ) può essere utilizzato per modificare alcune opzioni di stampa come ad esempio gli stili, i font, sebbene queste non possano essere salvate per il loro riutilizzo: le impostazioni verranno applicate esclusivamente al documento corrente.

Il pulsante di salvataggio non memorizza le opzioni generali di stampa, ma salva la configurazione attuale (**layout + DATI**), in un file che potrà essere riaperto in futuro, tramite l' icona "Load Report" ( ), per poter essere stampato.
N.B.: Ricaricando tale configurazione non vengono ripristinate le impostazioni, ma vengono caricati i dati presenti nel report al momento del salvataggio!

Invece, per poter modificare e rendere permanenti alcune opzioni di stampa possono essere definiti alcuni stili: dal menù File | Print Styles | Define Print Styles (o in alternativa attraverso pulsante della toolbar) si accede ad una maschera dalla quale sarà possibile definire uno o più stili di stampa con impostazioni ad-hoc riguardo a margini, formato del foglio, sfondi, intestazione, note a piè pagina ecc... Ad ognuno degli stili definiti potrà essere associato un nome identificativo, il quale comparirà come voce del menù File | Print Styles da dove potranno essere riutilizzati per eventuali stampe successive.

Infine attraverso le opzioni di Zoom si può impostare la visualizzazione della griglia:
- Fit to Page: Adatta l' intera griglia all' interno dei margini del foglio.
- Whole page: Adatta la griglia in modo che sia contenuta e visualizzata per intero all' interno della finestra di preview.
- Two, Four, Multiple Pages: Adatta la griglia in modo che sia contenuta e visualizzata su 2, 4 , più pagine, all' interno della finestra di

preview.

8.10.2Stampa gantt

Il gantt mette a disposizione un pannello di anteprima di stampa entro cui è possibile customizzare i parametri di stampa.

Le opzioni di stampa possono essere salvate su file per un successivo riutilizzo.

Nella toolbar sono presenti diversi pulsanti:

 per accedere al pannello delle opzioni di stampa in cui è possibile scegliere:

- se stampare solo la griglia o solo il gantt o entrambi
- quali margini di stampa utilizzare
- cosa stampare nell'header e nel footer delle pagine
- se limitare la stampa del gantt ad un orizzonte di stampa diverso da quello utilizzato dal report

Altri due pulsanti permettono di salvare su file o caricare da file le opzioni di stampa. Ci sono anche due pulsanti (obsoleti) per importare da file o esportare su file le opzioni di stampa.

Nel pannello di stampa è possibile:

- definire quale stampante utilizzare tra quelle installate nel sistema;
- decidere quali pagine stampare;
- avviare la stampa cliccando sul pulsante "Stampa";
- scalare la stampa in modo da rimpicciolirla o allargarla secondo una percentuale di zoom oppure cercare di adattare la stampa ad un certo numero di pagine verticali ed un certo numero di pagine orizzontali. Questo meccanismo di scalamento potrebbe non essere possibile nel caso in cui il report da stampare fosse molto grosso e il numero di pagine impostate esiguo, in questo caso l'applicazione mostrerà un messaggio di errore;
- ingrandire o rimpicciolire l'anteprima di stampa (questo zoom ha effetto solo sulla visualizzazione);

- navigare tra le pagine dell'anteprima di stampa.

8.11 Formule

Le formule sono utilizzate per definire condizioni e colonne o elementi calcolati.

Di seguito riportiamo l'elenco di operatori e funzioni, che il sistema mette a disposizione per definire condizioni necessarie alla realizzazione di filtri, decorazioni, colonne custom.

Operatori

() (parentesi)

****** (elevamento a potenza)

~ (inverso)

! (NOT)

***** (moltiplicazione)

/ (divisione)

% (resto)

%% (percentuale)

+ (addizione)

- (sottrazione)

< (minore)

<= (minore o uguale)

== (uguale)

<> != (diverso)

>= (maggiore o uguale)

> (maggiore)

| (OR)

^ (XOR)

& (AND)

Funzioni Stringa

TRIM(stringa1): elimina gli spazi all'inizio e alla fine di stringa1

LTRIM(stringa1): elimina gli spazi all'inizio di stringa1

RTRIM(stringa1): elimina gli spazi alla fine di stringa1

CONCATENATE(stringa1, stringa2 [, ..., stringaN]): concatena una serie di stringhe. Il numero di stringhe deve essere maggiore o uguale a 2

CONCAT(stringa1, stringa2 [, ..., stringaN]): equivalente a CONCATENATE

REPL (stringa1, n): concatena n copie di stringa1.
Risultato: una stringa costituita da n copie di stringa1 concatenate

LEFT(stringa1, n): ritorna una stringa contenente i primi n caratteri di stringa1

RIGHT(stringa1, n): ritorna una stringa contenente gli ultimi n caratteri di stringa1

SUBSTR(stringa1, indice, n): ritorna una stringa contenente n caratteri di stringa1 a partire dal carattere alla posizione indice. Indice parte da 1.
Esempio: SUBSTR('PIPPO', 2, 3) ritorna 'IPP'

TOSTR(espressione1): converte espressione1 in stringa

LEN(stringa1): Lunghezza di una stringa

EMPTY(stringa1): Verifica se una stringa è vuota
Risultato: Ritorna 1 se stringa1 è vuota, 0 altrimenti

COMPARE(stringa1, stringa2): Confronto tra 2 stringhe (CASE SENSITIVE).
Risultato: Ritorna 1 se le due stringhe sono uguali, 0 altrimenti

COMPARESTR(stringa1, stringa2): Confronto tra 2 stringhe (CASE SENSITIVE).
Risultato: Ritorna un numero minore di 0 se stringa1 è minore di stringa 2, uguale a 0 se le due stringhe sono uguali, maggiore di 0 se stringa1 è maggiore di stringa 2

COMPARETEXT(stringa1, stringa2): Confronto tra 2 stringhe.
Risultato: come COMPARESTR ma ignorando il maiuscolo/minuscolo dei caratteri.

STRINGTODATETIME(stringa, formato): Converte una stringa in una data/ora secondo il formato inserito.
Esempio: STRINGTODATETIME('28-10-2010', 'DD-MM-YYYY') ritorna il giorno 28/10/2010.

POS(substringa, stringa): cerca una stringa dentro un'altra.

Risultato: Ritorna la posizione di substringa dentro stringa. Se substringa non è presente ritorna 0. E' case sensitive.

Esempio:

POS('A', 'TABLE') ritorna 2.

UPPERCASE(stringa): converte stringa in caratteri maiuscoli.

LOWERCASE(stringa): converte stringa in caratteri minuscoli.

Funzioni Speciali

IF(condizione1, valore1, valore2): Valida la condizione.

Risultato: valore1 se condizione1 è vera, valore2 altrimenti

AND(espressione1, espressione2 [, ..., espressioneN]): AND logico tra le espressioni. Il numero di espressioni deve essere maggiore o uguale a 2.

Risultato: 0 se il risultato dell'AND logico è false, 1 se il risultato dell'AND logico è true

Esempio:

AND((FIELD1 > -0,2), (FIELD1 < 0,2)) == 0 ritorna true se il valore del campo FIELD1 è compreso tra -0,2 e +0,2

OR(espressione1, espressione2 [, ..., espressioneN]): OR logico tra le espressioni. Il numero di espressioni deve essere maggiore o uguale a 2.

Risultato: 0 se il risultato dell'OR logico è false, 1 se il risultato dell'OR logico è true

ROUND(espressione1, numero di decimali): arrotonda l'espressione 1 con il numero di decimali specificato
Esempio:
ROUND(8.761, 0) ritorna 8

ROUND(8.761, 1) ritorna 8.8
ROUND(8.761, 2) ritorna 8.76

NOT(espressione1): NOT logico dell'espressione
Risultato: 0 se l'espressione1 vale 1, 1 altrimenti

SUM(funzione di range) o **SUM**(elenco di valori): somma algebrica dei valori dei campi appartenenti al range o all'elenco

MAX(funzione di range) o **MAX**(elenco di valori): il valore massimo dei valori campi appartenenti al range o all'elenco

MIN(funzione di range) o **MIN**(elenco di valori): il valore minimo dei valori campi appartenenti al range o all'elenco

AVG(funzione di range) o **AVG**(elenco di valori): media algebrica dei valori dei campi appartenenti al range o all'elenco

COUNT(funzione di range) o **COUNT**(elenco di valori): numero dei campi appartenenti al range o all'elenco

TODAY(espressione1): distanza in giorni (valore intero) da oggi a espressione1

NOW(espressione1): distanza in giorni (valore float) dall'istante corrente a espressione1 (1=1 giorno, 0.5=12 ore, ecc..)

GETWEEK(espressione1): ritorna il numero della settimana dell'anno a cui appartiene espressione1

GETMONTH(espressione1): estrae il numero del mese da espressione1

GETYEAR(espressione1): estrae l'anno da espressione1

TODATE(anno, mese, giorno): converte 3 valori o espressioni in una data

TODATETIME(anno, mese, giorno, ora, minuti, secondi): converte 6 valori o espressioni in una data+ora

STARTOFTHEWEEK(espressione1): ritorna la data del lunedi della settimana a cui appartiene la data passata come espressione1

STARTOFTHEMONTH(espressione1): ritorna l'istante di inizio del mese a cui appartiene la data passata come espressione1

ENDOFTHEMONTH(espressione1): ritorna l'istante di fine del mese a cui appartiene la data passata come espressione1

Funzioni di Range

CHILDS(campo1): ritorna i valori di campo1 per tutti i figli del nodo corrente appartenenti al livello successivo. Deve essere usata sempre come parametro di una funzione SUM, MAX, MIN, COUNT o AVG.

Applicabilità: gantt e alberi

Esempio:

MAX(CHILDS(RITARDO))

CHILDSNOTNULL(campo1): ritorna i valori non nulli di campo1 per tutti i figli del nodo corrente appartenenti al livello successivo. Deve essere usata sempre come parametro di una funzione SUM, MAX, MIN, COUNT o AVG.

Applicabilità: gantt e alberi

PARENTS(campo1): ritorna il valore di campo1 per il nodo padre del nodo corrente. Deve essere usata sempre come parametro di una funzione SUM, MAX, MIN, COUNT o AVG.

Applicabilità: gantt e alberi

Esempio:

MIN(PARENTS(DATA_CONSEGNA))

PARENTSNOTNULL(campo1): ritorna il valore di campo1 per il nodo padre del nodo corrente se non è nullo. Deve essere usata sempre come parametro di una funzione SUM, MAX, MIN, COUNT o AVG.

Applicabilità: gantt

Esempio:

MIN(PARENTSNOTNULL(DATA_CONSEGNA))

Funzioni Costanti

NOW: Ritorna la data e l'ora correnti

TODAY: Ritorna la data corrente

Esempi di colonne calcolate

Su un gantt, se la fase è 10 scrivi 'ATTREZZAGGIO':

IF(NUMERO_FASE==10,'ATTREZZAGGIO','')

Su un gantt, se la fase è 20 e il codice risorsa è 13260 scrivi 'ATTENZIONE':

IF(AND((NUMERO_FASE==20),(COMPARETEXT(CODICE_RISORSA,'13260')==0)),'ATTENZIONE', '')

Su un albero, su di un nodo padre, calcolare la massima data di consegna dei nodi figli:

MAX(CHILDS(DATA_CONSEGNA))

Parte IX

9 Cubo

9.1 Operatività cubo

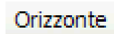
Il Cubo è un tipo di report che permette di effettuare una analisi multilivello del dato in due differenti modi:
Vista Indicatori: riporta una serie di indici di sintesi calcolati per un certo intervallo di tempo
Vista Cubo: visualizza (sia come numero, sia graficamente) come varia il dato nel tempo
Non tutte le informazioni che il cubo mette a disposizione possono essere utilizzate sia come indicatore sia come cubo; il numero di risorse sature, ad esempio, è un valore che ha significato solo come indice.
Un cubo può essere realizzato partendo da un predeterminato catalogo di indagine (ad esempio un layout delle risorse); gli elementi costituenti il catalogo (le risorse) sono detti Chiavi del cubo.
L'utilizzo di un catalogo non è comunque necessario, si possono combinare liberamente le chiavi di indagine (ad esempio Famiglia Prodotto/Codice Parte).

Toolbar Cubo

La toolbar cambia leggermente a seconda della vista attiva (Indicatori/Cubo).

 Per passare alla Vista Cubo (andamento valori nel tempo)

 Per passare alla Vista Indicatori



Per scegliere l'orizzonte temporale (tra quelli disponibili), per il quale visualizzare i dati



Per scegliere o definire nuovi filtri da effettuare sulle chiavi



Per scegliere lo scenario di visualizzazione tra quelli definiti, per modificare alcuni parametri della vista (vedi Modifica Viste), per scegliere il livello della chiave da visualizzare (ad esempio solo le foglie terminali di un catalogo gerarchico)

 Per ricercare un elemento della chiave

 Per definire filtri sui valori dei campi (Su quantità), indicando il valore minimo e massimo del range in cui deve essere contenuto; per definire filtri sulle chiavi (Su chiavi), indicando l'operatore e la stringa da cercare

 Per visualizzare la sezione eventualmente contenente tachimetri e istogrammi (se configurati)

 Per richiamare l'anteprima di stampa

 Visualizza i dati come valori numerici nel tempo

 Visualizza solo gli elementi chiave, selezionati tramite la box a sinistra del nome della chiave

 Esporta su Excell il dato, così come lo si sta visualizzando, cioè tenendo conto dell'esplosione o meno dei rami

Icone appartenenti solo alla toolbar della Vista Cubo:

 Per visualizzare il dato graficamente

 Minimizza tutte le righe del cubo (per ogni singola riga usare l'icona - in alto a destra)

 Ripristina tutte le righe del cubo (per ogni singola riga usare l'icona in alto a destra)

Vista Indicatori

Per ogni chiave del cubo, vengono riportati una serie di indicatori riferiti alla simulazione selezionata e ad un ben preciso intervallo di tempo (di default l'intervallo è l'intero orizzonte di simulazione).
Per i livelli non terminali, è possibile decidere che tipo di dato visualizzare (somma, massimo, media etc dei valori dei livelli figli) (vedi Modifica Viste).
Cliccando sull'intestazione di una colonna si esegue l'ordinamento per il campo in oggetto.

Menù Tasto Destro

Posizionati su una cella degli indici (o sulla colonna Chiave), con il tasto destro del mouse si accede al seguente menù:
Vai a: si passa alla Vista Cubo (andamento nel tempo), per l'elemento chiave selezionato
Configura: vedi Configurazione Indicatori
Espandi ramo (da colonna chiave): espande il ramo selezionato
Autosize: ridimensiona la larghezza delle colonne in funzione del testo in esse contenuto
Visualizza dettaglio buckets: voce che se attivata, per la colonna selezionata, visualizza i dati di tale campo nel tempo (ove non è possibile, viene riportata l'informazione globale sul primo bucket)
Strumenti – Collega a: per aggiungere, all'interno di un layout, report collegati al cubo
Strumenti – Esegui: permette di accedere a funzioni di modifica del dato, diverse in relazione al cubo origine
Strumenti – Rinfresca report dipendenti: per aggiornare all'elemento selezionato, tutti gli eventuali report collegati al cubo

Configurazione Indicatori

La voce di menù Configura richiama una finestra per la configurazione grafica degli indici.

Si può definire l'aspetto grafico della colonna selezionata e alcuni parametri generali della tabella.

Parametri Colonna

<u>Valori numerici</u>

Per definire:

- il formato numerico con il quale visualizzare il valore
- l'eventuale unità di misura da associare vicino al valore
- il colore dello sfondo
- la larghezza della colonna
- l'allineamento e il font del carattere
- se visualizzare o meno il valore 0
- per quali livelli visualizzare il dato in oggetto (esempio se si imposta >2, non verrà visualizzato il valore per i primi 2 livelli di aggregazione)

<u>Intestazione Colonna</u>

Per definire:

- il colore dello sfondo dell'intestazione della colonna
- se usare gli stessi colori già definiti per la visione grafica del cubo
- il font della intestazione

Tools Grafici Colonna

Per definire:

- il tipo di visualizzazione del dato tra valore numerico, istogramma e istogramma con fondo scala scelto tra i campi resi disponibili dal cubo
- due soglie di allarme (alta e bassa), in funzione delle quali colorare font o istogramma o aggiungere icone
- se e in che modo considerare le condizioni di colorazione già impostate per la vista cubo

Parametri Generali Tabella

Per definire:

- il colore dello sfondo delle righe pari e dispari
- l'altezza delle righe
- il colore dello sfondo della colonna chiave
- se visualizzare la chiave in rilievo
- se visualizzare il check vicino alla chiave (e quindi poter selezionare gli elementi per essere filtrati)

Vista Cubo

Per ogni chiave del cubo, viene riportato l'andamento nel tempo delle grandezze oggetto di analisi.

I dati possono essere visualizzati in formato numerico o grafico ( ).

Nel caso di visualizzazione grafica, la scala delle grandezze può essere diversa per ogni elemento chiave, in quanto dipende dal massimo

valore raggiunto da una grandezza nel periodo considerato.
Su ogni singolo elemento chiave compaiono le seguenti icone.
In alto a sinistra

 Porta l'elemento in cima alla lista e lo tiene fisso durante lo scorrimento verticale

 Permette il passaggio da valore numerico a grafico per il solo elemento

 Presente se il ramo è esploso; ogni puntino identifica un livello superiore e permette cliccandolo di chiudere il ramo al livello voluto

In alto a destra

 Isola elemento, minimizza tutti gli elementi meno quello selezionato

 Minimizza / Ripristina l'elemento

Legenda

Per ogni elemento chiave, esiste una legenda riportante il nome della grandezza graficata e il colore con cui viene rappresentata nel grafico.
Posizionandosi su uno dei campi, con il tasto destro del mouse si può decidere come rappresentarlo graficamente:

- Tipo grafico: nessun grafico, linea, a gradino, istogramma, area (none, line, step, bar, area)
- Spessore linea: nel caso di linea o gradino, si indica lo spessore (valori da 1 a 4)

9.2 Modifica viste

Dalla toolbar principale del cubo, si accede a questo pannello dal quale si può creare una nuova vista o modificare quelle esistenti.

Parametri generali vista
Tipo: per definire se all'apertura del cubo, la vista cubo debba essere grafica o numerica
Nulli: cosa visualizzare in caso di dati nulli (niente o zero)
Frame: colore dello sfondo delle riga di intestazione
H. Testo: altezza delle righe per la visualizzazione numerica
H. Grafico: altezza delle righe per la visualizzazione grafica
Filtro: possibilità di associare di default uno dei vincoli preconfigurati
Mostra somme: la possibilità di visualizzare in fondo a destra per ogni elemento, nel caso di vista numerica, la somma per ogni grandezza

Dettagli vista
Campo: nome della grandezza (non modificabile)

Descrizione: nome con cui la grandezza viene visualizzata (editabile)
Dato: si definisce se visualizzare la grandezza nel tempo (vista cubo)
Indicatore: si definisce se visualizzare la grandezza come indicatore (vista indicatori)
Tipo dato: in caso di chiavi a più livelli, si definisce quale valore indicare per gli aggregati (somma, massimo, minimo …)
Tipo grafico: definisce la rappresentazione grafica (nessun grafico, linea, a gradino, istogramma, area)
Sfondo: definisce il metodo di colorazione dello sfondo della cella, nel caso sia verificata la condizione definita nelle Formule (vedi dopo)
Colore: colore con cui viene graficata la grandezza
Linea: spessore della linea, in caso di grafico a linea o a gradino
Formule: si possono definire delle condizioni di colorazione, indicando le formule e i colori da utilizzare

Parte X

10 Report di analisi di AtomosVision

AtomosVision permette di creare dei report di analisi costituiti da catene di report legati da vincoli di master-detail (padre-figlio).
I report sono alimentati da query o da tabelle in memoria attraverso cui viene estratta l'informazione che sarà poi usata per popolare griglie, cubi, pivot e quant'altro.

10.1 Report master

I report master possono essere alimentati:

- da query
- da tabella in memoria
- da un provider basato su una funzione (script rops)

In caso di alimentazione da query, l'utente può scegliere una delle query definite nel sistema oppure può inserirne una nuova attraverso il Query Manager.
In caso di alimentazione da provider, l'utente può scegliere uno dei provider (statico o dinamico master) definiti nel sistema dall'apposito pannello.

10.1.1 Report master da una query o una tabella in memoria

Questo è il caso più semplice di report master.
Il report (griglia, pivot, ecc..) è alimentato da una singola query o da una tabella in memoria. I record estratti dalla query o dalla tabella sono utilizzati per popolare il report prescelto.

10.1.2 Report master da due query

Vengono utilizzate due query quando si desidera creare un report master contenente informazione gerarchica, come ad esempio una distinta parti.

Lo scopo quindi di questo tipo di report è estrarre da db le informazioni relative ad una gerarchia di valori e di farlo appoggiandosi a due query.

La prima delle query servirà per estrarre le informazioni relative alla gerarchia, cioè i legami della gerarchia.

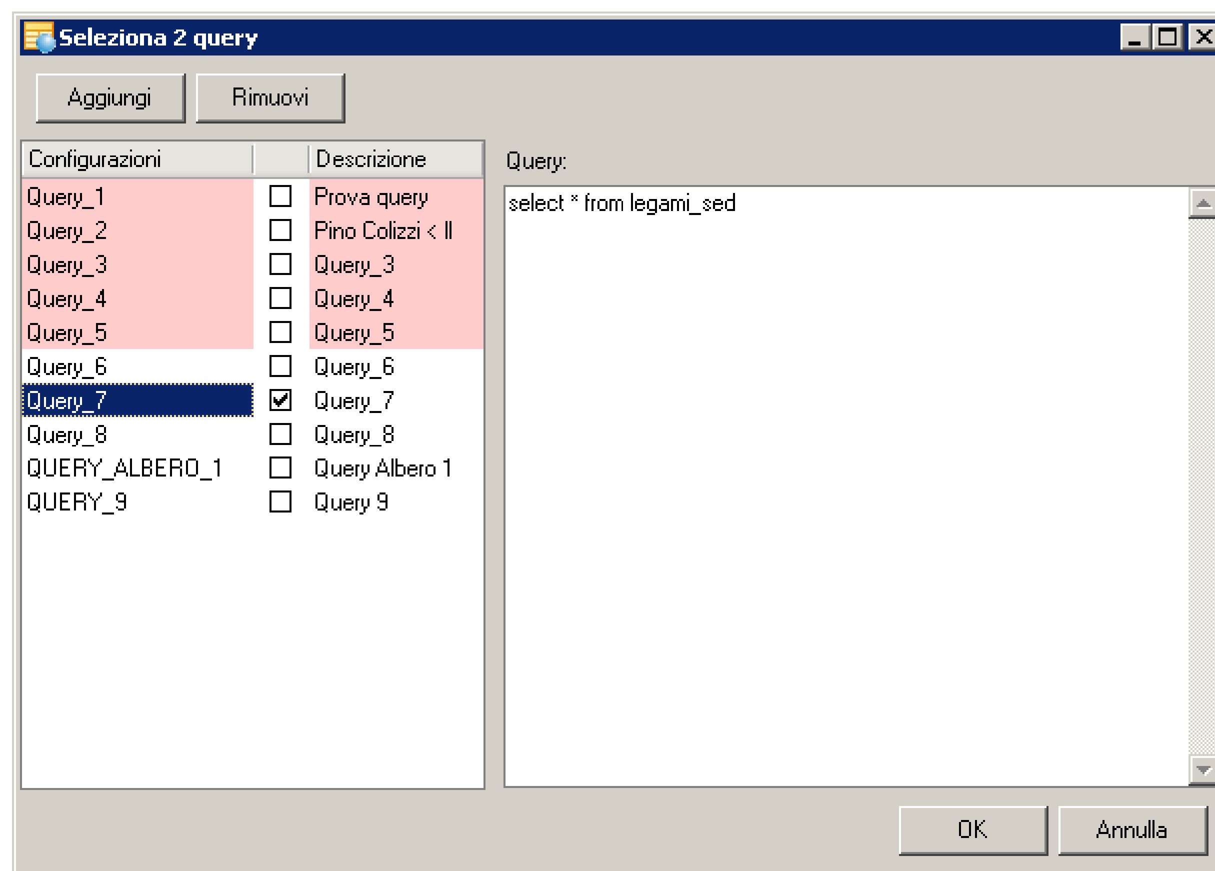
La seconda query servirà per estrarre le informazioni relative ai singoli nodi della gerarchia.

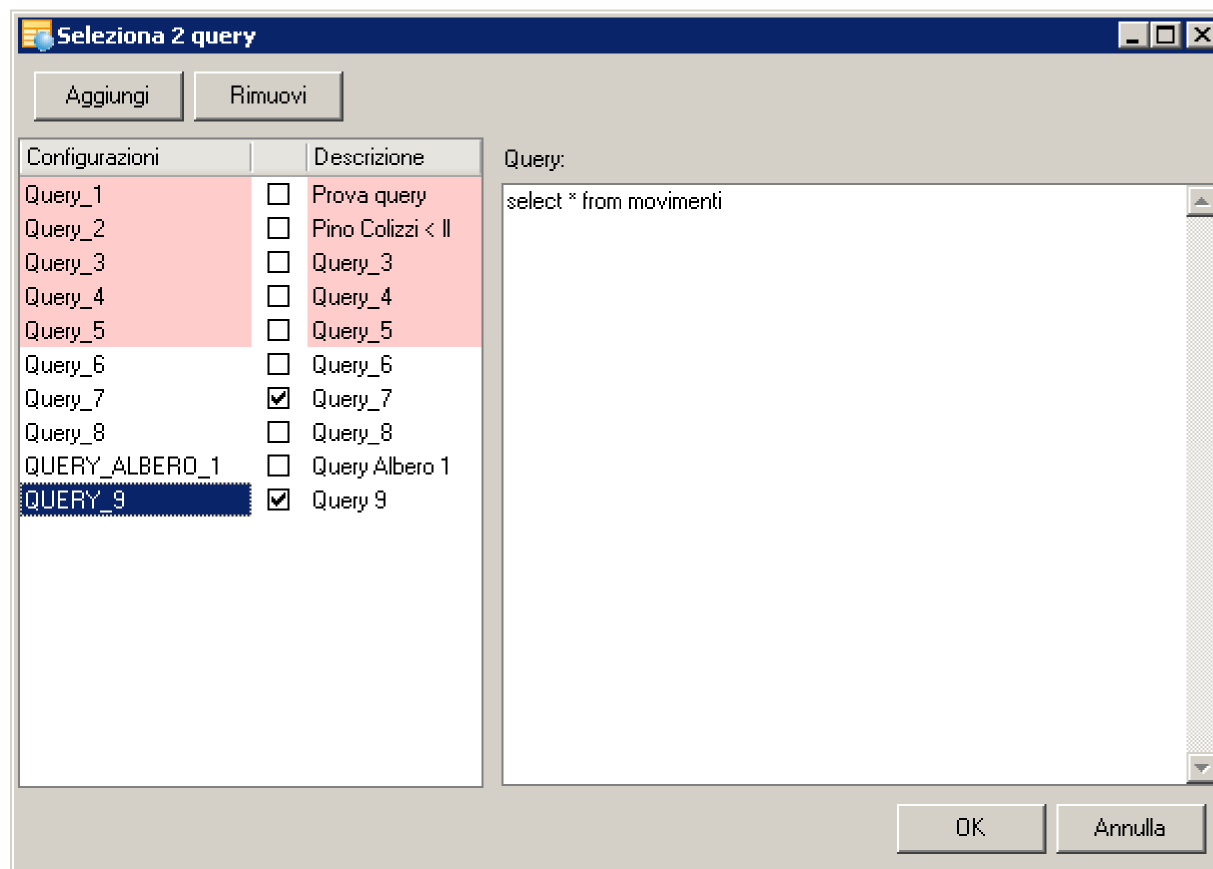
Vediamo un esempio ritagliato sul database di schedulazione Nicim:

vogliamo costruire un report gerarchico (albero o gantt) per visualizzare i legami generati dallo schedulatore.

I legami sono salvati dallo schedulatore nella tabella LEGAMI_SED, le informazioni relative ai nodi della gerarchia possono essere invece essere estratti dalla tabella MOVIMENTI.

Inseriamo e selezioniamo 2 query come in figura:


Seleziona 2 query
Aggiungi
Rimuovi
Configurazioni
Descrizione
Query_1 Prova query
Query_2 Pino Colizzi < ll
Query_3 Query_3
Query_4 Query_4
Query_5 Query_5
Query_6 Query_6
Query_7 Query_7
Query_8 Query_8
QUERY_ALBERO_1 Query Albero 1
QUERY_9 Query 9
Query:
select * from legami_sed
OK
Annulla



Nella maschera di costruzione della gerarchia selezioniamo Query_7 per i legami e Query_9 per i nodi:

Costruzione della gerarchia

Tabella legami
Query_7
CODICE_ORDINE_FIGLIO
CODICE_ORDINE_PADRE
CODICE_PARTE_FIGLIO
CODICE_VERSIONE_FIGLIO
DATA_ORA_PREL_FIGLIO
DATA_ORA_VERS_FIGLIO
ID_ERRORE
MAG_LOGICO
NATURA_FIGLIO
NATURA_PADRE
PROGR
QTA_COPERTA_PADRE
QTA_PREL_FIGLIO
QTA_PREL_SCOPERTA

Campi chiave padri:

Campi chiave figli:

Tabella nodi
QUERY_9
ALTERNATIVA
CAUSALE_MOVIMENTO
CHNG_BY
CLASSE_ALTERN
CLIENTE_FORNITORE
COD_RAGGR
CODICE_ASSIEME
CODICE_COMMESSA
CODICE_FASE
CODICE_MAGAZZINO

Campi chiave:

OK Annulla

e poi col drag'n'drop configuriamo i campi come in figura:

Costruzione della gerarchia

Tabella legami

Query_7

CODICE_PARTE_FIGLIO
CODICE_VERSIONE_FIGLIO
DATA_ORA_PREL_FIGLIO
DATA_ORA_VERS_FIGLIO
ID_ERRORE
MAG_LOGICO
PROGR
QTA_COPERTA_PADRE
QTA_PREL_FIGLIO
QTA_PREL_SCOPERTA
QTA_VERS_FIGLIO
QTA_VERS_SCOPERTA
REPARTO
SIM

Campi chiave padri:

CODICE_ORDINE_PADRE
NATURA_PADRE

Campi chiave figli:

CODICE_ORDINE_FIGLIO
NATURA_FIGLIO

Tabella nodi

QUERY_9

ALTERNATIVA
CHNG_BY
CLASSE_ALTERN
CLIENTE_FORNITORE
COD_RAGGR
CODICE_ASSIEME
CODICE_COMMESSA
CODICE_FASE
CODICE_MAGAZZINO
CODICE_PARTE

Campi chiave:

CODICE_ORDINE
CAUSALE_MOVIMENTO

OK Annulla

Abbiamo così definito 3 doppiette di campi che consentono di identificare chiaramente:

- chi è figlio/padre di chi
- quale informazione è associata ad ogni nodo

E' importante che il numero di campi per ogni gruppo di chiavi sia lo stesso e che essi combacino per tipo e significato: nel nostro

esempio NATURA_PADRE, NATURA_FIGLIO e CAUSALE_MOVIMENTO sono campi dello stesso tipo e tutti e 3 identificano la natura di un movimento/ordine.

Il report verrà quindi riempito sulla base dei dati estratti da database.

10.1.3 Report master implosione/esplosione

Un altro modo per costruire un report master che contiene una gerarchia è quello di definire una regola di discesa come implosione/esplosione sui legami di una gerarchia.

Questa configurazione è realizzata inserendo 3 query distinte:

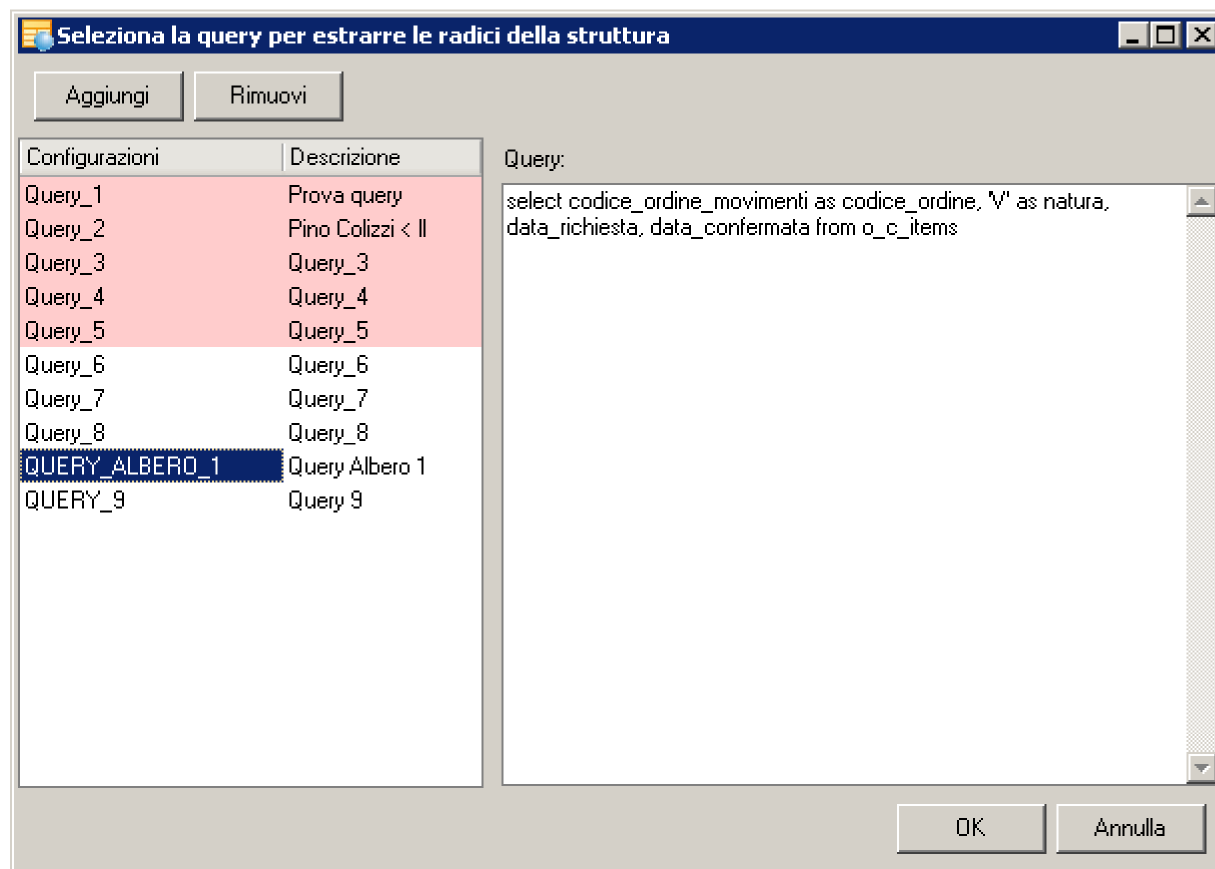
- la prima query serve ad estrarre le radici o la radice della gerarchia
- la seconda query serve per trovare i nodi figli a partire da un nodo padre
- la terza query serve per estrarre da database le informazioni inerente i singoli nodi

Vediamo un esempio ritagliato sul database di schedulazione Nicim:

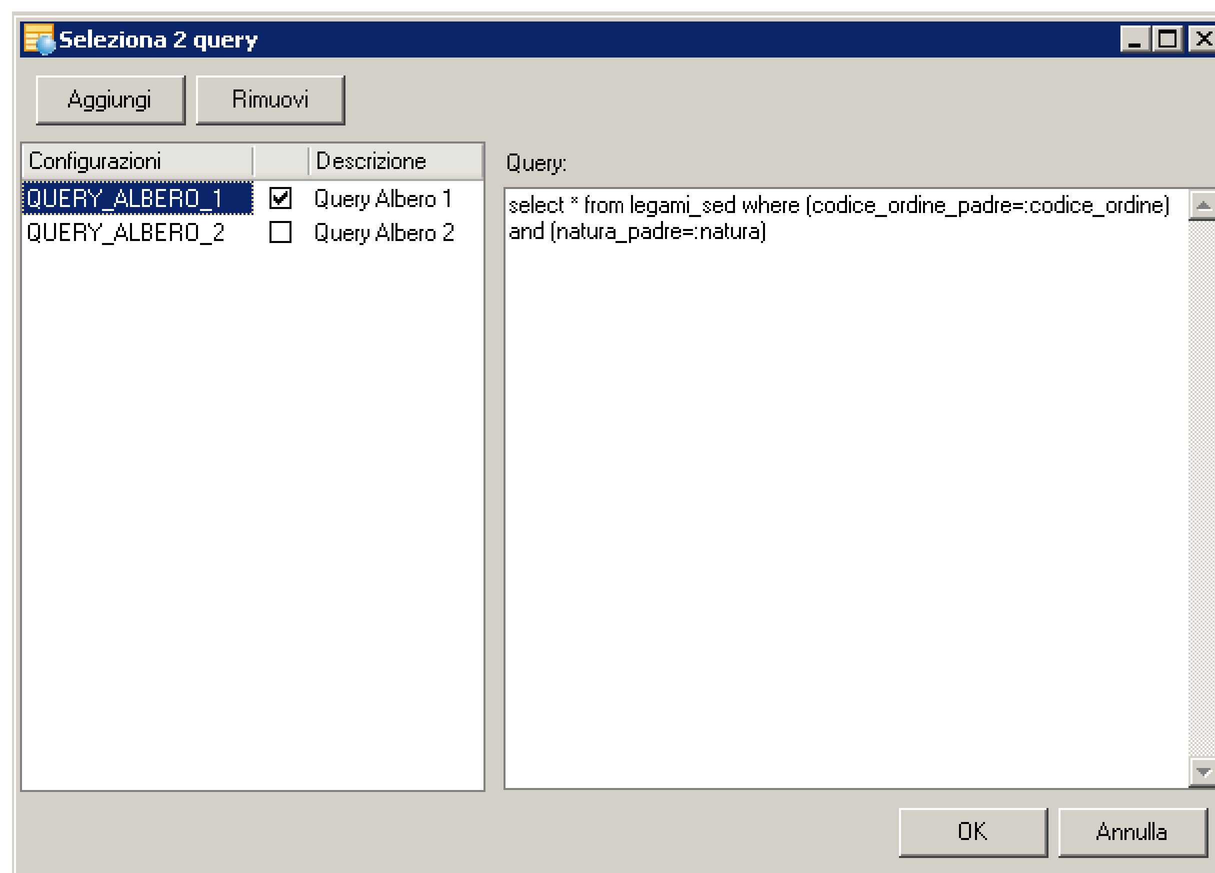
vogliamo costruire un report gerarchico (albero o gantt) per visualizzare i legami generati dallo schedulatore.

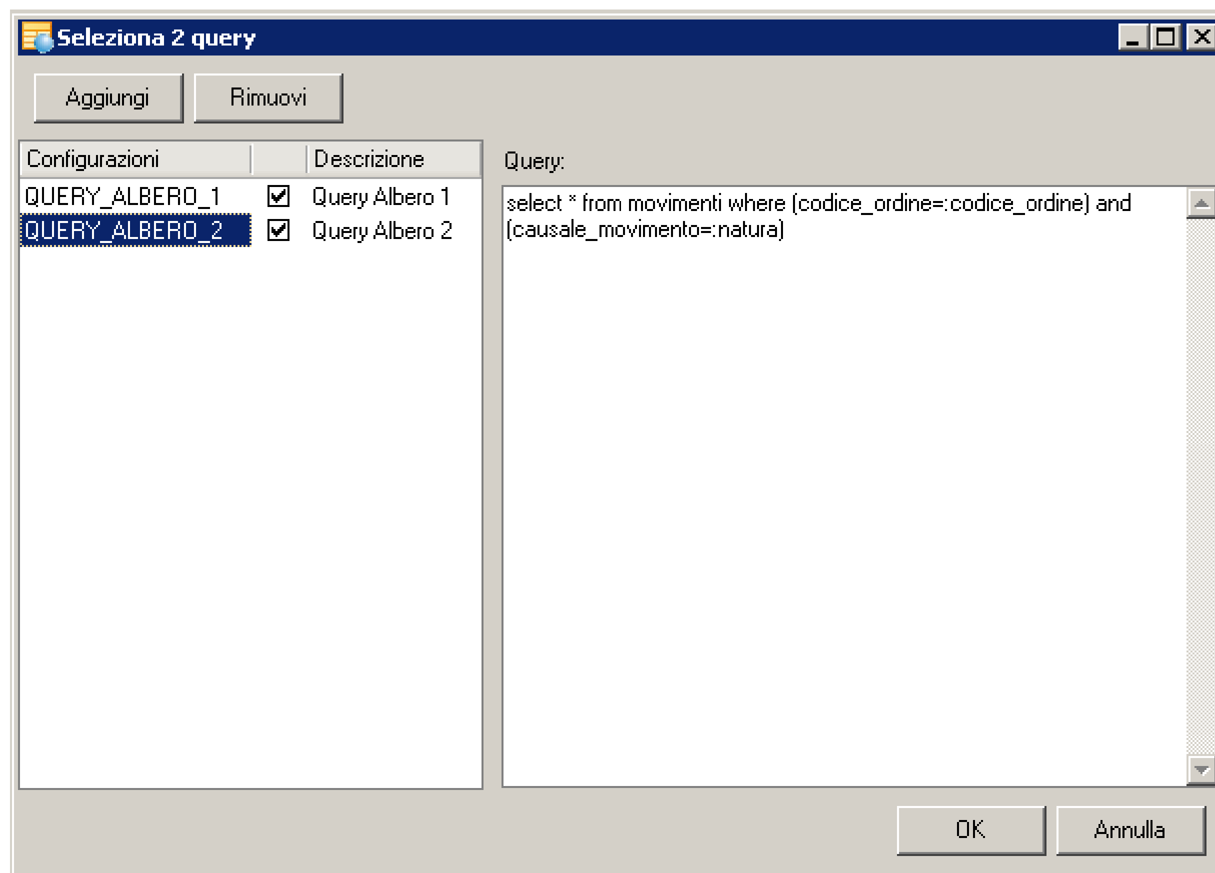
I legami sono salvati dallo schedulatore nella tabella LEGAMI_SED, le informazioni relative ai nodi della gerarchia possono essere invece essere estratti dalla tabella MOVIMENTI, l'elenco delle radici (cioè gli ordini cliente) è contenuto nella tabella O_C_ITEMS.

Inseriamo e selezioniamo la query per estrarre le radici come in figura:



Inseriamo e selezioniamo la query per esplodere la gerarchia e quella per estrarre le informazioni sui nodi come in figura:


Seleziona 2 query
Aggiungi
Rimuovi
Configurazioni
Descrizione
QUERY_ALBERO_1
Query Albero 1
QUERY_ALBERO_2
Query Albero 2
Query:
select * from legami_sed where (codice_ordine_padre=:codice_ordine)
and (natura_padre=:natura)
OK
Annulla



Queste due query meritano una particolare attenzione perchè non sono delle semplici select. E' necessario inserire anche una where condition contenente i parametri necessari per legare:

- la query dei legami con il nodo padre
- la query delle informazioni sul nodo con il nodo corrente

Nel nostro caso ogni nodo della gerarchia dei LEGAMI_SED è identificato univocamente dal suo codice e dalla sua natura.

Ora configuriamo la gerarchia definendo per ogni query il suo ruolo, come in figura:

Costruzione della gerarchia

Query radici
QUERY_ALBERO_1
CODICE_ORDINE
DATA_CONFERMATA
DATA_RICHIESTA
NATURA

Campi chiave:

Query legami
QUERY_ALBERO_1
codice_ordine
natura

Campi chiave padri:

CODICE_ORDINE_FIGLIO
CODICE_ORDINE_PADRE
CODICE_PARTE_FIGLIO
CODICE_VERSIONE_FIGLIO
DATA_ORA_PREL_FIGLIO
DATA_ORA_VERS_FIGLIO
ID_ERRORE
MAG_LOGICO
NATURA_FIGLIO

Campi chiave figli:

Query nodi
QUERY_ALBERO_2
codice_ordine
natura

Campi chiave:

OK Annulla

e poi col drag'n'drop configuriamo le diverse chiavi necessarie a definire l'implosione/esplosione (nel nostro caso è una esplosione):

Costruzione della gerarchia

Query radici
QUERY_ALBERO_1
DATA_CONFERMATA
DATA_RICHIESTA

Campi chiave:
CODICE_ORDINE
NATURA

Query legami
QUERY_ALBERO_1

Campi chiave padri:
codice_ordine
natura

CODICE_ORDINE_PADRE
CODICE_PARTE_FIGLIO
CODICE_VERSIONE_FIGLIO
DATA_ORA_PREL_FIGLIO
DATA_ORA_VERS_FIGLIO
ID_ERRORE
MAG_LOGICO
NATURA_PADRE
PROGR

Campi chiave figli:
CODICE_ORDINE_FIGLIO
NATURA_FIGLIO

Query nodi
QUERY_ALBERO_2

Campi chiave:
codice_ordine
natura

OK Annulla

E' importante osservare come sia necessario preservare una simmetria tra i vari gruppi di chiavi (stesso numero e tipologia) ed il ruolo che hanno avuto i parametri che abbiamo dovuto inserire nelle query al punto precedente: per la query di esplosione dei legami la chiave dei nodi padre è proprio il set di parametri inseriti, così come per la query di estrazione delle informazioni relative ai singoli nodi.

Se avessimo voluto costruire una implosione al posto della esplosione saremmo potuti partire da una query capace di estrarre le giacenze di magazzino invece che le testate degli ordini, poi la query dei legami avrebbe dovuto sfruttare i campi CODICE_ORDINE_PADRE e NATURA_PADRE invece che gli omologhi per i nodi figli mentre la query dei nodi avrebbe potuto essere riutilizzata senza modifiche.

10.2 Report detail

I report detail possono essere alimentati:

- da una o più query
- da una tabella in memoria
- direttamente dai dati del dataset del report master
- da un provider basato su una funzione (script rops)

In caso di alimentazione da query, l'utente può scegliere una delle query definite nel sistema oppure può inserirne una nuova attraverso il Detail Query Manager.

10.2.1 Report detail da una query o una tabella in memoria

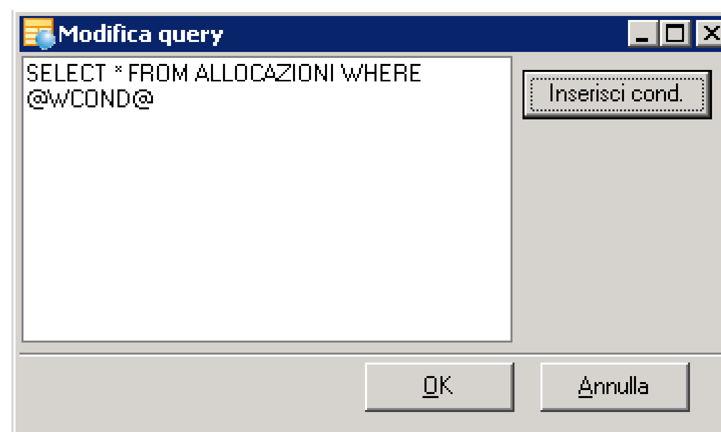
Report detail da query

Per creare un nuovo report detail da query:

- cliccare su "Aggiungi"
- assegnare un nome univoco alla query di master-detail (il nome non potrà più essere modificato perchè funzionerà da chiave per la

query)

- cliccare su "Edita query"
- inserire la query di dettaglio. Nella WHERE condition inserire, magari in AND con altre condizioni o come condizione unica, il tag che rappresenta la where condition automatica generata dall'applicazione per legare il report master col detail. Il tag viene inserito cliccando sul pulsante "Inserisci cond."



- cliccare su "Aggiungi" per aggiungere tutte le condizioni di join necessarie. Una condizione può essere rimossa col pulsante "Elimina". Per ogni condizione è necessario selezionare una coppia di campi: il primo tra quelli della query di dettaglio ("Campo query di dettaglio"), il secondo ("Valore") tra quelli del report master.

- In "Campo originale" si deve lasciare il valore inserito automaticamente dall'applicativo a meno che nella query non sia stato definito un ALIAS (AS) su un campo utilizzato in una condizione. Se è definito un alias, l'applicazione ha bisogno che l'utente inserisca a mano il valore del nome del campo in "Campo originale" altrimenti non sarà in grado di risolvere la query di dettaglio.

Alcuni aspetti tecnici più avanzati sono discussi in questo parametro.

Report detail da tabella in memoria

Per creare un nuovo report detail da query:

- cliccare su "Aggiungi"
- assegnare un nome univoco alla configurazione di master-detail (il nome non potrà più essere modificato perchè funzionerà da chiave per la configurazione)
- selezionare una tabella in memoria nell'elenco delle tabelle estratte dal CDB
- cliccare su "Aggiungi" per aggiungere tutte le condizioni di join necessarie. Una condizione può essere rimossa col pulsante "Elimina". Per ogni condizione è necessario selezionare una coppia di campi: il primo tra quelli della tabella di dettaglio ("Campo tabella di dettaglio"), il secondo ("Valore") tra quelli del report master.

10.2.1.1 Ottimizzazione delle query di detail

Alcune considerazioni tecniche per ottenere collegamenti master-detail piu' efficienti.

Quando Vision deve risolvere una query di detail (tipicamente perchè l'utente seleziona una o più celle/righe sul report master) provvede a generare una o più stringhe sql e ad eseguire una o più query verso il db.

La stringa SQL del detail viene generata ciclando sulle righe selezionate sul report master e sulle condizioni di join.

Se ad esempio ho 2 condizioni di join: `CODICE_PARTE->CODICE_PARTE` e `CODICE_VERSIONE->CODICE_VERSIONE`, e ho 2 righe selezionate sul report master: `('PARTE_A', 'VERSIONE_A')` e `('PARTE_B', 'VERSIONE_B')` la query SQL per estrarre i dati di detail assomiglierà a qualcosa di questo tipo:

```
SELECT qualcosa FROM tabella WHERE ((CODICE_PARTE=:CODICE_PARTE1) AND (CODICE_VERSIONE=:CODICE_VERSIONE1)) OR ((CODICE_PARTE=:CODICE_PARTE2)
```

e i parametri verranno così sostituiti:

```
CODICE_PARTE1 = 'PARTE_A'
CODICE_VERSIONE1='VERSIONE_A'
CODICE_PARTE2='PARTE_B'
CODICE_VERSIONE2='VERSIONE_B'
```

Quindi:

- più celle/righe sono selezionate sul report master (attenzione che un grand-total su una pivot può essere equivalente a centinaia di dati atomici) e più la query di detail diventa lunga e più parametri vengono annegati nella query
- più condizioni di join sono configurate e più la query di detail diventa lunga e più parametri vengono annegati nella query
- più i nomi dei campi sono lunghi e più la query di detail diventa lunga

La lunghezza massima di una stringa SQL ha però dei limiti che dipendono dal DB (ad esempio 64Kb per Oracle) e anche il numero di parametri può avere un tetto massimo (ad esempio 2100 per MS SQLServer http://msdn.microsoft.com/en-us/library/ms143432.aspx). Quando Vision si rende conto che generando la query di detail si rischia di superare uno di questi limiti, **se la query di detail non contiene clausole GROUP BY**, cerca di generare più query di detail per mantenere la stringa SQL più corta o per limitare il numero di parametri, le esegue in parallelo e poi fa il distinct dei risultati delle varie query.

Tutto ciò ovviamente può rallentare il refresh del report detail. Per questo può essere utile cercare di ottimizzare, usando il buon senso e tenendo presente quello qui illustrato, la query e la configurazione di detail durante la creazione del report.

10.2.2 Report detail implosione/esplosione

E' possibile costruire un report detail come implosione/esplosione di una gerarchia di legami a partire da un set di radici.
Il set di radici è estratto con una query di detail a partire dalla selezione di un report master.

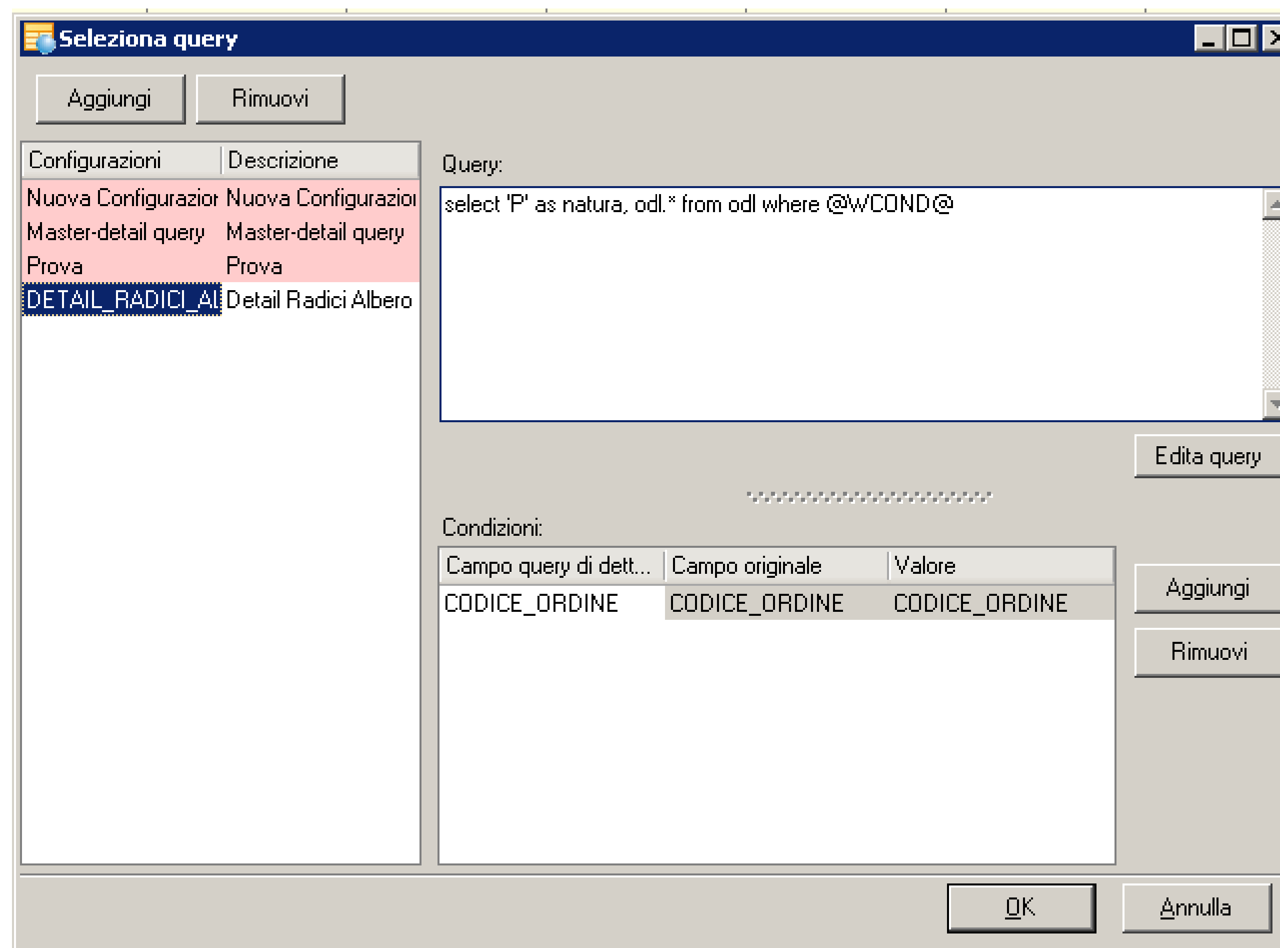
Analogamente al report master implosione/esplosione, anche questo report è costruito a partire da 3 query distinte:

- la prima query serve ad estrarre le radici o la radice della gerarchia e, **diversamente dal report master, è una query di detail risolta a partire dalle righe selezionate sul report master**
- la seconda query serve per trovare i nodi figli a partire da un nodo padre
- la terza query serve per estrarre da database le informazioni inerente i singoli nodi

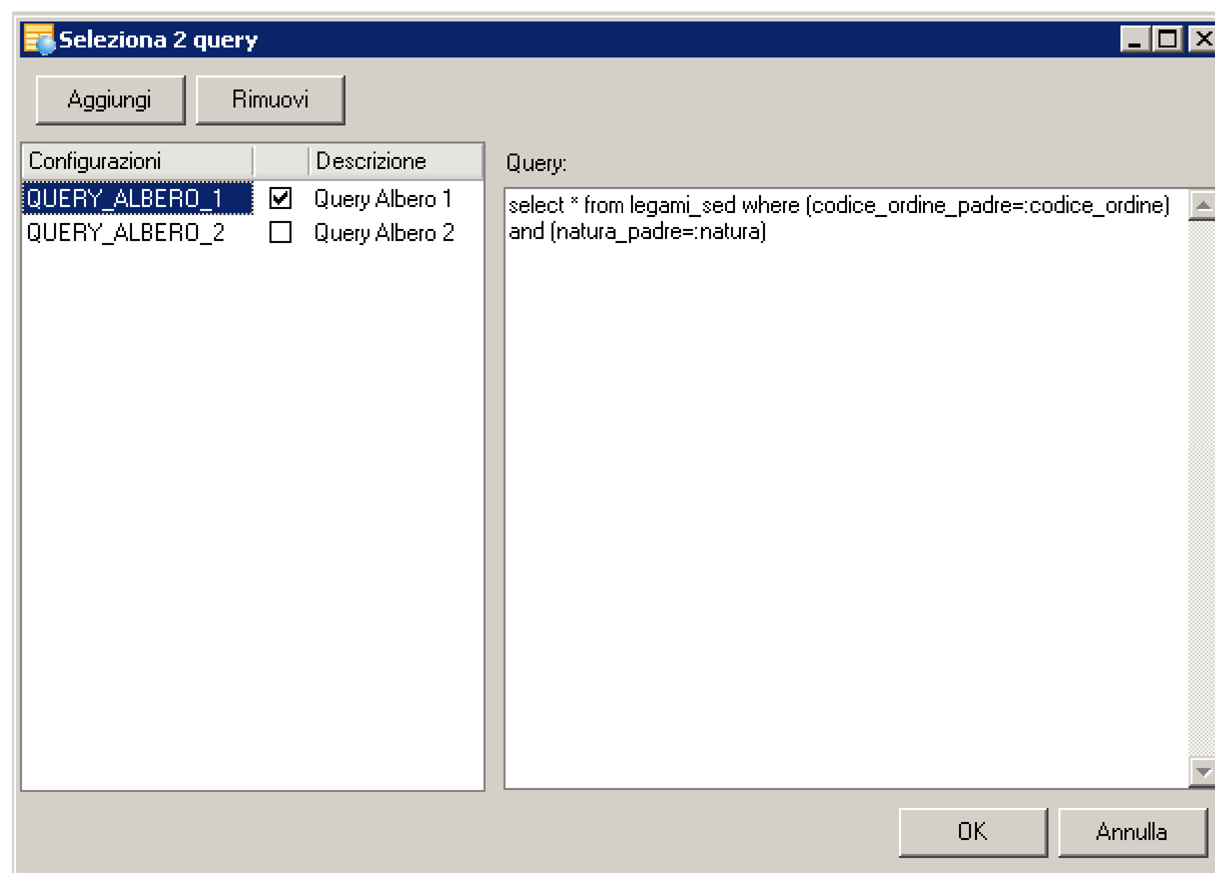
La procedura di configurazione è analoga a quella del report master implosione/esplosione e a quella si rimanda.
L'unica differenza è che la selezione della prima query, quella per estrarre le radici, deve essere effettuata tra le query di detail inserite nel sistema (nel report master invece sono disponibile tutte le query master del Query Manager).

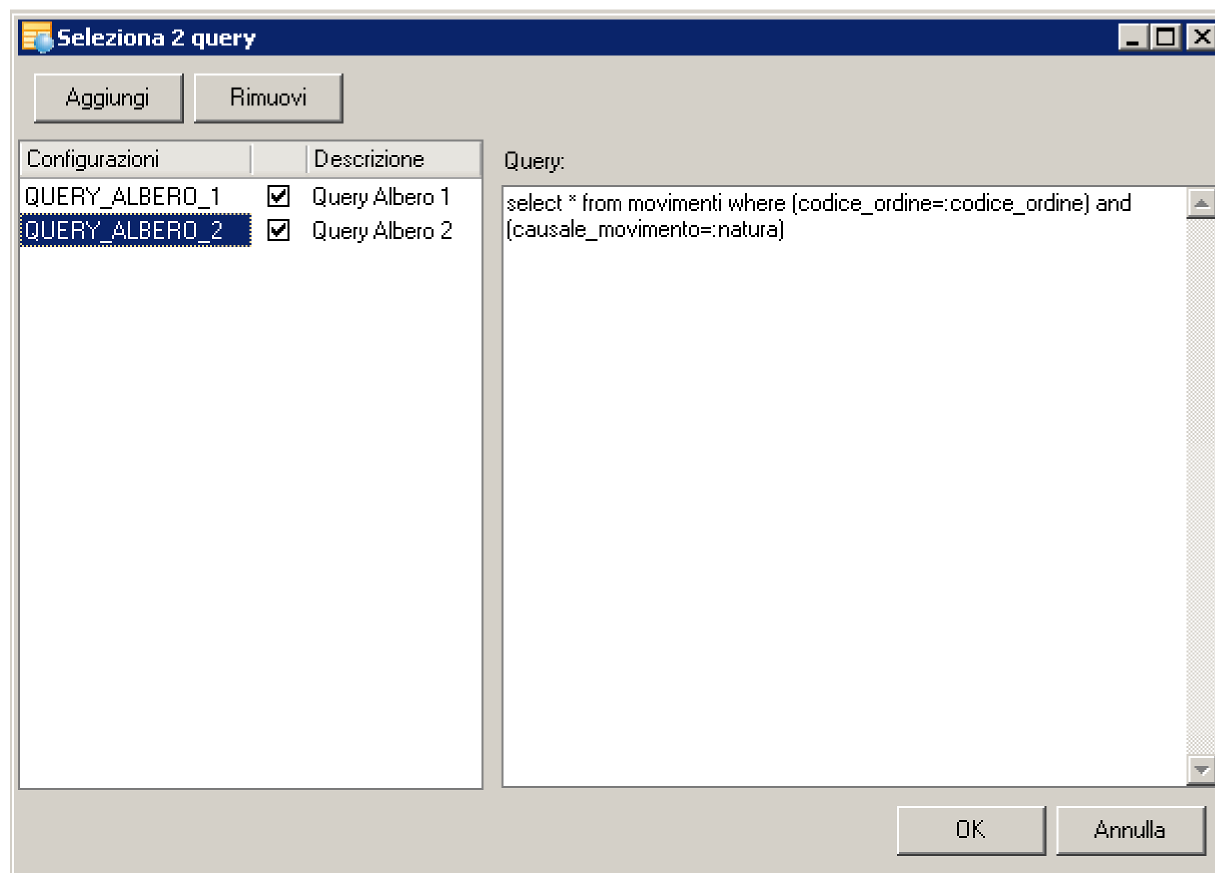
Vediamo un esempio ritagliato sul database di schedulazione Nicim:

vogliamo costruire un report gerarchico (albero o gantt) per visualizzare i legami generati dallo schedulatore a partire dagli ODL selezionati in un report master (ad esempio una griglia).
I legami sono salvati dallo schedulatore nella tabella LEGAMI_SED, le informazioni relative ai nodi della gerarchia possono essere invece essere estratti dalla tabella MOVIMENTI, l'elenco delle radici (cioè gli ODL) è contenuto nella tabella ODL.
Inseriamo e selezioniamo la query per estrarre le radici come in figura:



Inseriamo e selezioniamo la query per esplodere la gerarchia e quella per estrarre le informazioni sui nodi come in figura:


Seleziona 2 query
Aggiungi
Rimuovi
Configurazioni
Descrizione
QUERY_ALBERO_1
Query Albero 1
QUERY_ALBERO_2
Query Albero 2
Query:
select * from legami_sed where (codice_ordine_padre=:codice_ordine) and (natura_padre=:natura)
OK
Annulla



Queste due query meritano una particolare attenzione perchè non sono delle semplici select. E' necessario inserire anche una where condition contenente i parametri necessari per legare:

- la query dei legami con il nodo padre
- la query delle informazioni sul nodo con il nodo corrente

Nel nostro caso ogni nodo della gerarchia dei LEGAMI_SED è identificato univocamente dal suo codice e dalla sua natura.

Ora configuriamo la gerarchia definendo per ogni query il suo ruolo, come in figura:

Costruzione della gerarchia

Query radici
Detail Radici Albero
AVANTI
CLASSE
CLIENTE
COD_RAGGR

Campi chiave:

Query legami
QUERY_ALBERO_1
codice_ordine
natura

CODICE_ORDINE_FIGLIO
CODICE_ORDINE_PADRE
CODICE_PARTE_FIGLIO
CODICE_VERSIONE_FIGLIO
DATA_ORA_PREL_FIGLIO
DATA_ORA_VERS_FIGLIO
ID_ERRORE

Campi chiave padri:

Campi chiave figli:

Query nodi
QUERY_ALBERO_2
codice_ordine
natura

Campi chiave:

OK
Annulla

e poi col drag'n'drop configuriamo le diverse chiavi necessarie a definire l'implosione/esplosione (nel nostro caso è una esplosione):

Costruzione della gerarchia

Query radici: Detail Radici Albero
- MOV_MODO_PREL
- MOV_MODO_VERS
- NO_ANTICIPO
- NUMERO_ORDINE

Campi chiave:
- CODICE_ORDINE
- NATURA

Query legami: QUERY_ALBERO_1

Campi chiave padri:
- codice_ordine
- natura

- ID_ERRORE
- MAG_LOGICO
- NATURA_PADRE
- PROGR
- QTA_COPERTA_PADRE
- QTA_PREL_FIGLIO
- QTA_PREL_SCOPERTA

Campi chiave figli:
- CODICE_ORDINE_FIGLIO
- NATURA_FIGLIO

Query nodi: QUERY_ALBERO_2

Campi chiave:
- codice_ordine
- natura

OK | Annulla

E' importante osservare come sia necessario preservare una simmetria tra i vari gruppi di chiavi (stesso numero e tipologia) ed il ruolo che hanno avuto i parametri che abbiamo dovuto inserire nelle query al punto precedente: per la query di esplosione dei legami la chiave dei nodi padre è proprio il set di parametri inseriti, così come per la query di estrazione delle informazioni relative ai singoli nodi.

Se avessimo voluto costruire una implosione al posto della esplosione la query dei legami avrebbe dovuto sfruttare i campi CODICE_ORDINE_PADRE e NATURA_PADRE invece che gli omologhi per i nodi figli mentre la query dei nodi avrebbe potuto essere riutilizzata senza modifiche.

10.2.3 Report detail da dataset del report master

E' possibile costruire un report detail alimentato direttamente dai dati selezionati su di un report master.
E' utile quando il report master offre una rappresentazione del dato originario aggregata o elaborata, come avviene sui report pivot, cubo e grafico.
Nell'esempio è utilizzato per mostrare quali valori del dataset originario contribuiscono al valore di una cella grand total di una pivot:

10.3 Pivot sincronizzata

La pivot sincronizzata è un tipo particolare di pivot che nasce dall'esigenza di poter visualizzare all'interno dello stesso report di analisi due o più pivot alimentate da sorgenti dati differenti ma che condividono la stessa gerarchia di livelli.

Nell' esempio l'utente vuole visualizzare il piano vendite per modello-marca e il fatturato per venditore (sempre per modello-marca). La gerarchia delle macchine è la medesima per entrambi i report:

Report di analisi

Pivot sincronizzata da query

Scenario

Drop Filter Fields Here

Id | Data

Marca	Modello	mag 2008	giu 2008	lug 2008	Grand Total
FIAT	500			2	2
	GRANDE PUNTO	1	1	1	3
FIAT Total		1	1	3	5
FORD	FIESTA	1	1	1	3
	KA	1	1		2
FORD Total		2	2	1	5
OPEL	CORSA	1	1		2

Pivot sincronizzata da query

Scenario

Drop Filter Fields Here

Prezzo | Nome

Marca	Modello	DANILO	MARIO	PIERO	Grand Total
FIAT	500	12000,00		10000,00	22000,00
	GRANDE PUNTO	26500,00	25000,00	26800,00	78300,00
FIAT Total		38500,00	25000,00	36800,00	100300,00
FORD	FIESTA		40000,00	14000,00	54000,00
	KA		12500,00	18800,00	31300,00
FORD Total			52500,00	32800,00	85300,00
OPEL	CORSA			30000,00	30000,00

Minimizzando la larghezza delle colonne delle Row header area della seconda pivot si ottiene una vista compatta della due pivot:

Drop Filter Fields Here

Id | Data

Marca	Modello	mag 2008	giu 2008	lug 2008	Grand Total
FIAT	500			2	2
	GRANDE PUNTO	1	1	1	3
FIAT Total		1	1	3	5
FORD	FIESTA	1	1	1	3
	KA	1	1		2
FORD Total		2	2	1	5
OPEL	CORSA	1	1		2

Drop Filter Fields Here

Nome

DANILO	MARIO	PIERO	Grand Total
12000,00		10000,00	22000,00
26500,00	25000,00	26800,00	78300,00
38500,00	25000,00	36800,00	100300,00
	40000,00	14000,00	54000,00
	12500,00	18800,00	31300,00
	52500,00	32800,00	85300,00

Tutte le pivot sincronizzate costruite all'interno del medesimo report di analisi hanno la caratteristica di mantenersi sincronizzate durante

lo scorrimento verticale dei dati (attraverso la tastiera, la barra di scorrimento o la rotellina del mouse), di mantenere sincronizzato lo stato espanso/collassato dei nodi che costituiscono la gerarchia della pivot e di mantenere sincronizzato l'ordinamento dei campi visualizzati.

Parte XI

11 FastReport

FastReport è uno strumento di reportistica integrato dentro AtomosVision che permette di creare i report di stampa.
I report di FastReport non sono navigabili ma possono essere stampati o pubblicati come documenti PDF, OpenOffice ODF, Word DOC, ecc...

Per il corretto funzionamento dei report di stampa è necessario che su Windows sia installata una stampante predefinita!

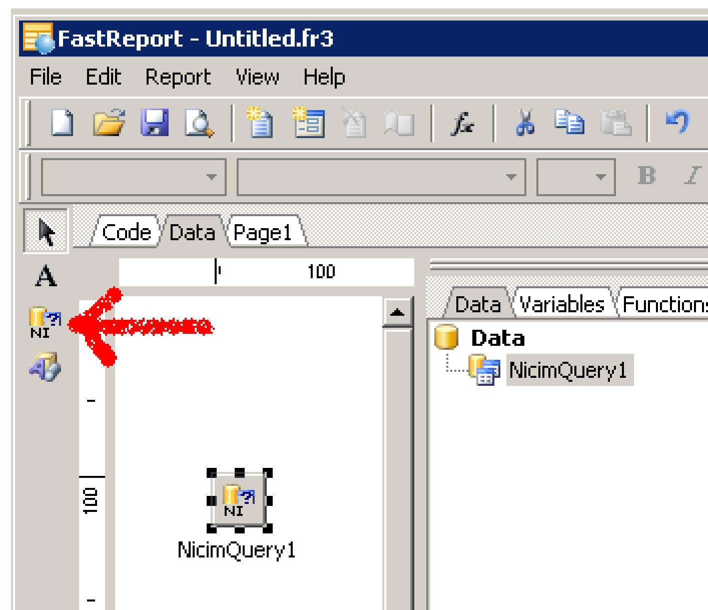
Il manuale utente e dello sviluppatore di FastReport possono essere scaricati come file help o pdf a questo indirizzo: http://fast-report.com/en/download/fast-report-4-download.html

Il manuale utente di FastReport è anche lanciabile da qui. Nota: i service pack recenti di Microsoft Windows (ad esempio Windows XP SP2 e recenti) bloccano i link a file esterni su dischi di rete all'interno degli help (file con estensione .chm) impedendo di fatto la corretta visione del file di help sui sistemi più aggiornati. Per ovviare al problema si può provare a ricopiare il file .chm in locale e a lanciarlo a mano.

FastReport integrato in AtomosVision non utilizza i componenti standard per l'accesso al db ma utilizza dei componenti customizzati.

11.1 Accesso ai dati

FastReport integrato in AtomosVision rende disponibile un componente custom per l'accesso ai dati sul db: TFrxNicimQuery.



Il funzionamento è analogo a quello dei componenti TFrx***Query nativi di FastReport e descritti nella documentazione di FastReport (Data Access Components) tranne che per quanto concerne la connessione al database.
La TFrxNicimQuery infatti si appoggia automaticamente alla connessione al db in uso in AtomosVision e non ha bisogno ne' di alcun componente extra (TFrx***DataBase) ne' di settaggi specifici per funzionare.

11.2 Funzioni di PascalScript

FastReport contiene al suo interno un ambiente di sviluppo con cui integrare script e macro nei report.
Il linguaggio utilizzato di default è il PascalScript.

Questo è l'elenco delle funzioni disponibili:

- function IntToStr(i: Integer): String
- function FloatToStr(e: Extended): String
- function DateToStr(e: Extended): String
- function TimeToStr(e: Extended): String
- function DateTimeToStr(e: Extended): String
- function VarToStr(v: Variant): String
- function StrToInt(s: String): Integer
- function StrToInt64(s: String): Integer
- function StrToFloat(s: String): Extended
- function StrToDate(s: String): Extended
- function StrToTime(s: String): Extended
- function StrToDateTime(s: String): Extended

- function Format(Fmt: String; Args: array): String
- function FormatFloat(Fmt: String; Value: Extended): String
- function FormatDateTime(Fmt: String; DateTime: TDateTime): String
- function FormatMaskText(EditMask: string; Value: string): string

- function EncodeDate(Year, Month, Day: Word): TDateTime
- procedure DecodeDate(Date: TDateTime; var Year, Month, Day: Word)
- function EncodeTime(Hour, Min, Sec, MSec: Word): TDateTime
- procedure DecodeTime(Time: TDateTime; var Hour, Min, Sec, MSec: Word)

- function Date: TDateTime
- function Time: TDateTime
- function Now: TDateTime
- function DayOfWeek(aDate: TDateTime): Integer
- function IsLeapYear(Year: Word): Boolean
- function DaysInMonth(nYear, nMonth: Integer): Integer

- function Length(s: Variant): Integer
- function Copy(s: String; from, count: Integer): String
- function Pos(substr, s: String): Integer
- procedure Delete(var s: String; from, count: Integer)
- procedure DeleteStr(var s: String; from, count: Integer)
- procedure Insert(s: String; var s2: String; pos: Integer)
- function Uppercase(s: String): String
- function Lowercase(s: String): String
- function Trim(s: String): String
- function NameCase(s: String): String
- function CompareText(s, s1: String): Integer
- function Chr(i: Integer): Char
- function Ord(ch: Char): Integer
- procedure SetLength(var S: Variant; L: Integer)

- function Round(e: Extended): Integer

- `function Trunc(e: Extended): Integer`
- `function Int(e: Extended): Integer`
- `function Frac(X: Extended): Extended`
- `function Sqrt(e: Extended): Extended`
- `function Abs(e: Extended): Extended`
- `function Sin(e: Extended): Extended`
- `function Cos(e: Extended): Extended`
- `function ArcTan(X: Extended): Extended`
- `function Tan(X: Extended): Extended`
- `function Exp(X: Extended): Extended`
- `function Ln(X: Extended): Extended`
- `function Pi: Extended`

- `procedure Inc(var i: Integer; incr: Integer = 1)`
- `procedure Dec(var i: Integer; decr: Integer = 1)`
- `procedure RaiseException(Param: String)`
- `procedure ShowMessage(Msg: Variant)`
- `procedure Randomize`
- `function Random: Extended`
- `function ValidInt(cInt: String): Boolean`
- `function ValidFloat(cFlt: String): Boolean`
- `function ValidDate(cDate: String): Boolean`

- `function CreateOleObject(ClassName: String): Variant`

- `function VarArrayCreate(Bounds: Array; Typ: Integer): Variant`
- `function VarType(V: Variant): Integer`

Valori restituiti da VarType:

```
varEmpty
varNull
varSmallint
varInteger
varSingle
varDouble
varCurrency
varDate
varOleStr
varDispatch
varError
varBoolean
varVariant
varUnknown
varShortInt
varByte
varWord
varLongWord
```

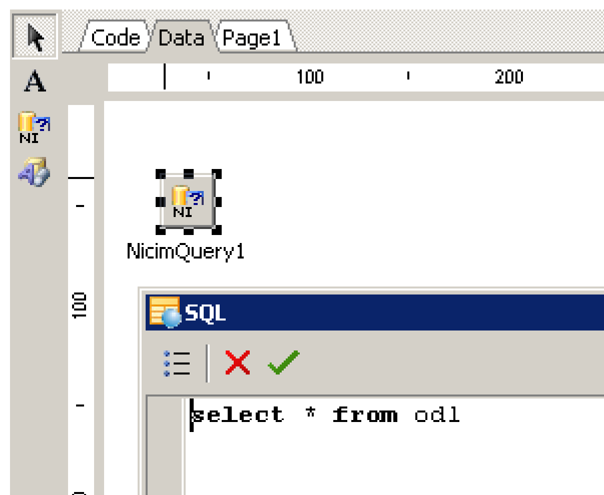
```
varInt64
varStrArg
varString
varAny
varTypeMask
varArray
varByRef
```

11.3 Come creare un report master-detail

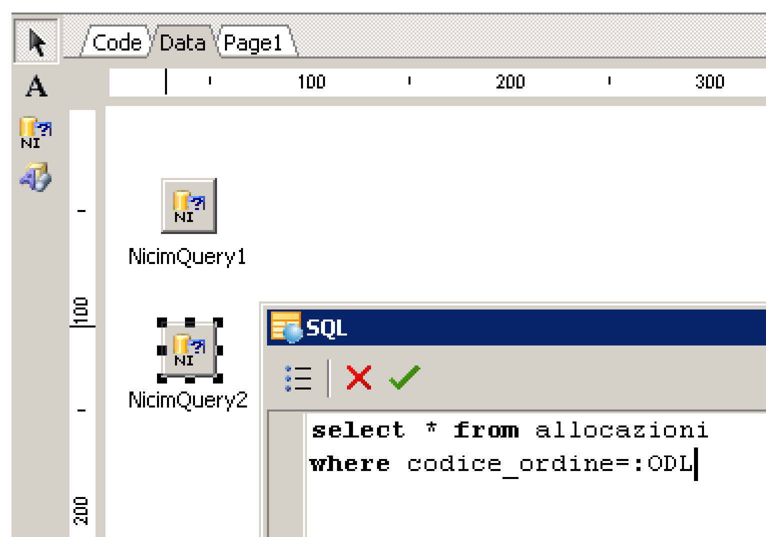
Vediamo come creare un report basato su una query master e una query di dettaglio.

In questo esempio vogliamo esplodere le fasi allocate a partire dagli ODL presenti sul database.

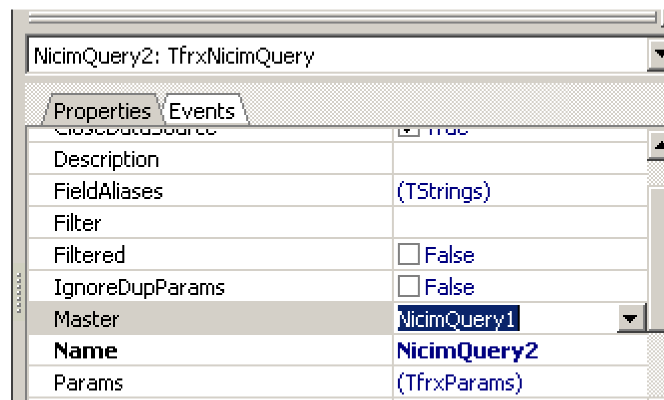
Creiamo una nuova query nella pagina Data del report in questo modo:



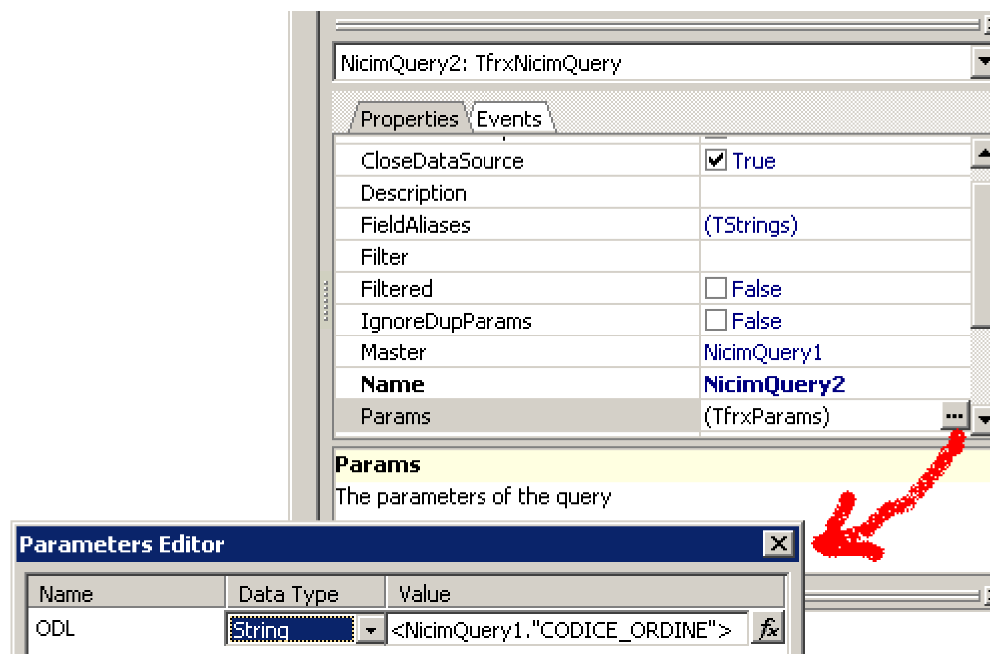
Creiamo una seconda query che costituirà la query di dettaglio per estrarre le fasi allocate a partire da un certo ODL:



Nelle proprietà di NicimQuery2 modifichiamo la proprietà Master:

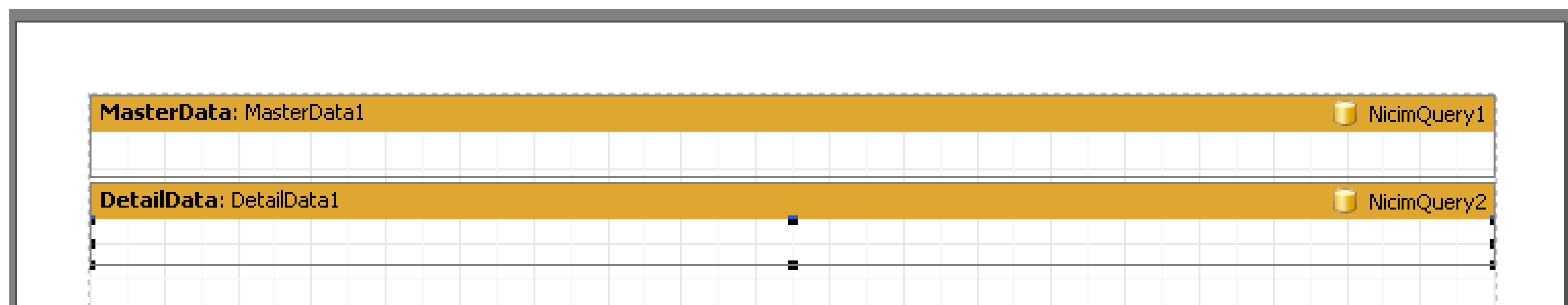


Poi selezioniamo la proprietà Params e modifichiamola in questo modo:

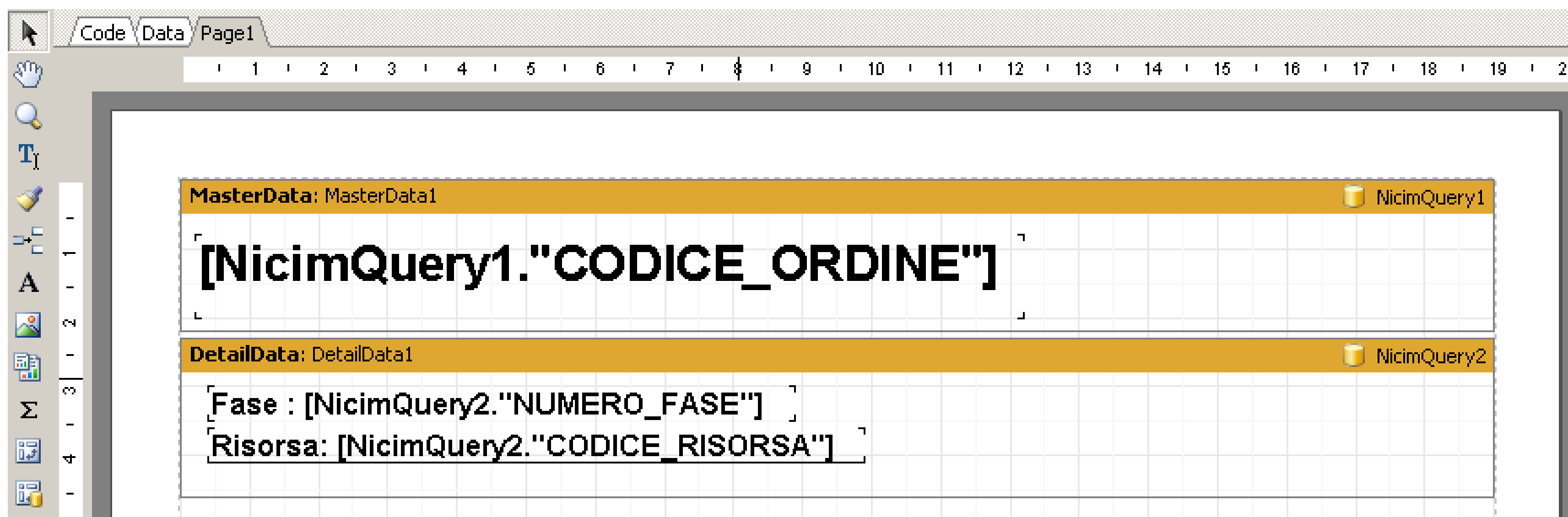


In questo modo colleghiamo il parametro ODL della query di dettaglio al campo appropriato (CODICE_ORDINE) della query master.

Nel report aggiungiamo un'area MasterData ed un'area DetailData e colleghiamole alle due query appena inserite:

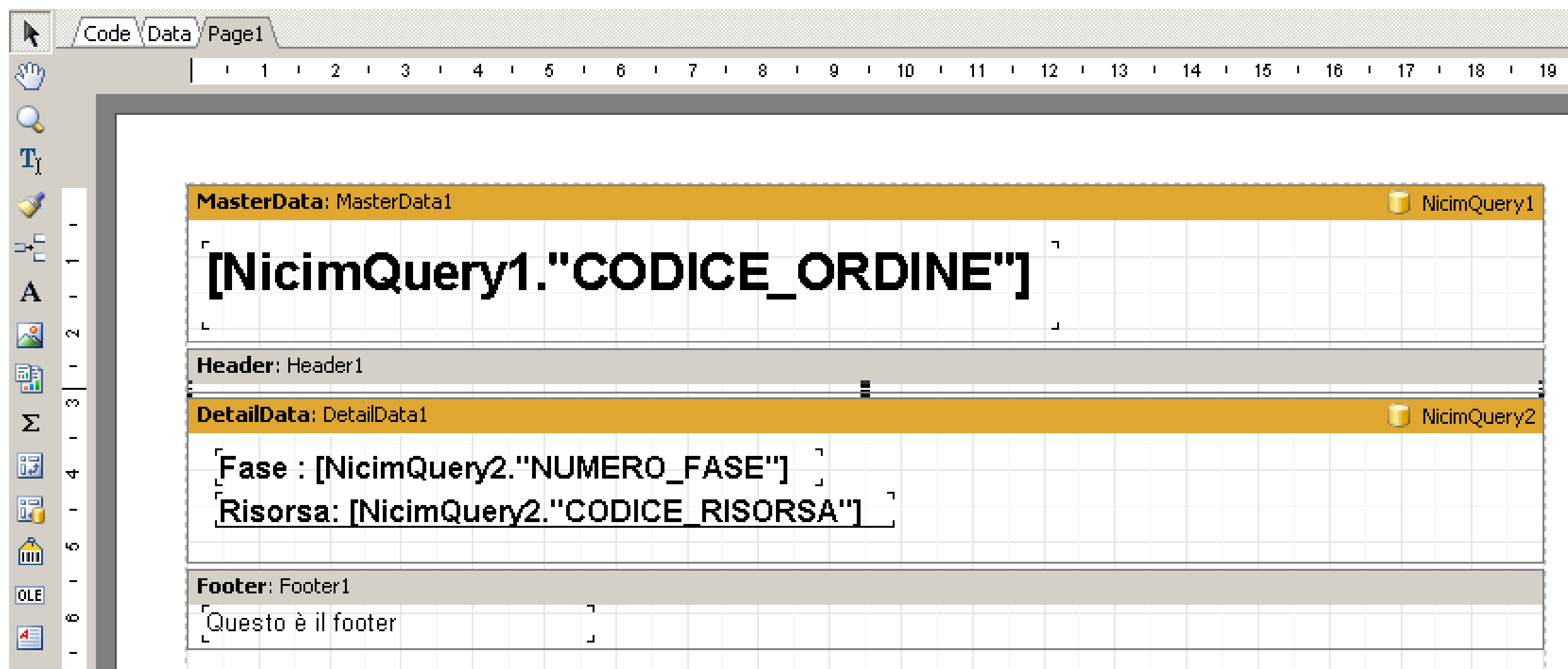


Poi disponiamo i campi desiderati (a scelta) nelle due aree:

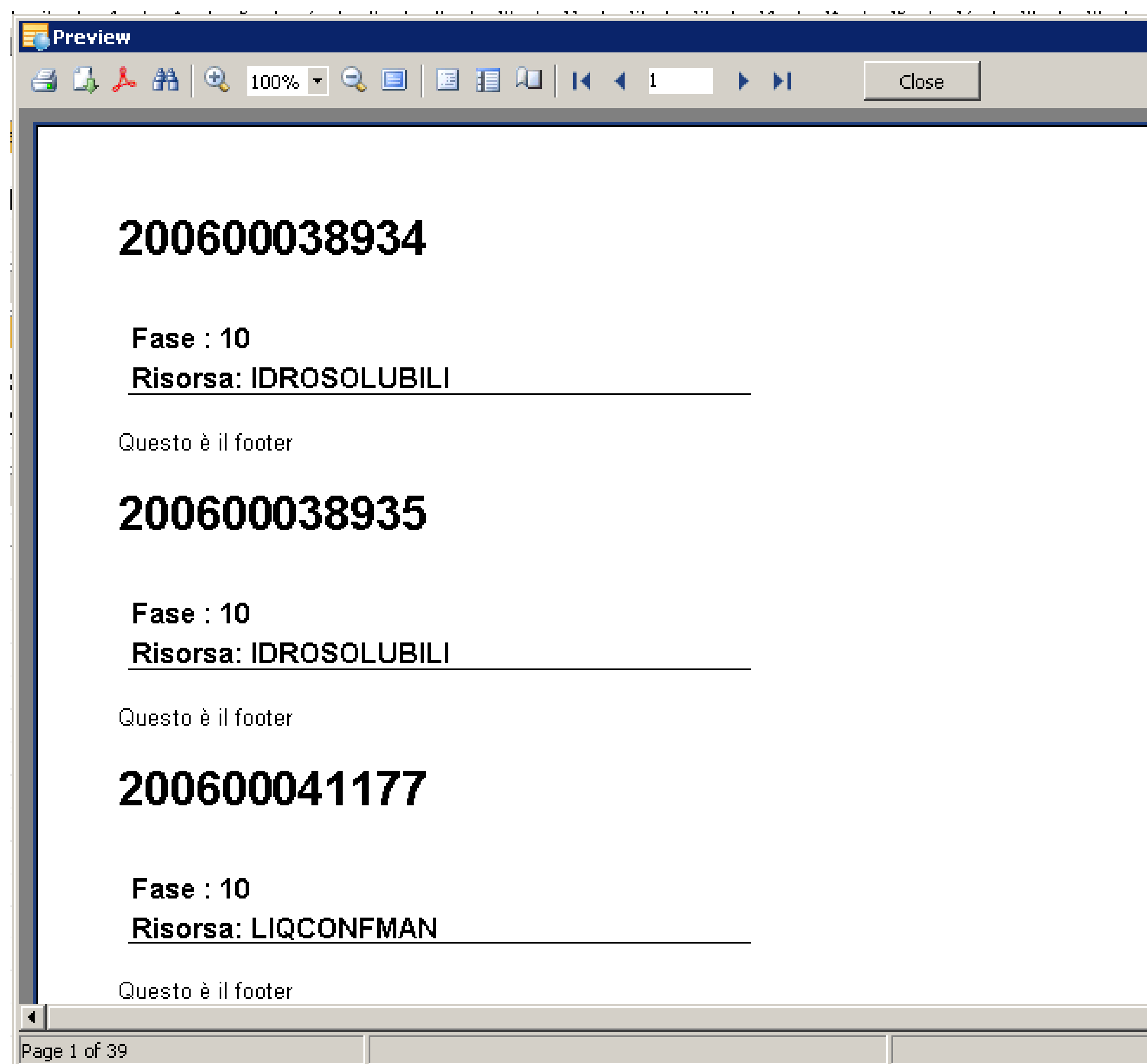


Il report è ora pronto per essere visualizzato.

Eventualmente è possibile aggiungere un'area Header e un'area Footer a cavallo della DetailData ad esempio per fornire una legenda alle informazioni di dettaglio:



Questo è il risultato finale:


Preview
100%
1
Close
200600038934
Fase : 10
Risorsa: IDROSOLUBILI
Questo è il footer
200600038935
Fase : 10
Risorsa: IDROSOLUBILI
Questo è il footer
200600041177
Fase : 10
Risorsa: LIQCONFMAN
Questo è il footer
Page 1 of 39

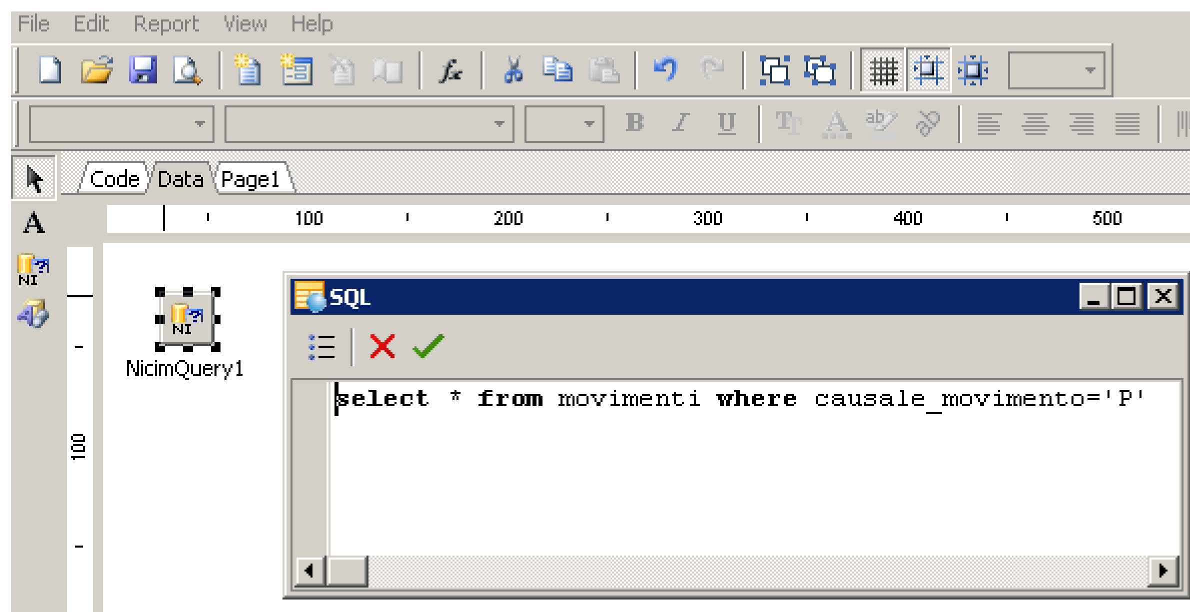
Attenzione: **se la query di detail non restituisse alcun dato, anche la parte master non verrebbe stampata**.
Per abilitare la stampa dell'intestazione master anche quando la query detail è vuota è necessario selezionare l'area MasterData collegata alla query master e nel pannello dei parametri associati alla MasterData selezionare l'opzione `PrintIfDetailEmpty` ed impostarne il valore a `True`.

11.4 Come creare una pivot con un campo percentuale calcolato tra le colonne

Vediamo come creare una pivot contenente una colonna che esprime il rapporto tra altre 2 colonne come percentuale.

A partire dai dati della tabella MOVIMENTI di Nicim, vogliamo creare una pivot che raggruppi per CODICE_PARTE tutte le quantità degli ODL e tutte le quantità consuntivate e vogliamo aggiungere alla pivot una colonna che mostri il rapporto tra QUANTITA e QUANTITA_CONSUNTIVO (quantità consuntivata) come percentuale.

In un nuovo report di stampa aggiungiamo una query che estragga i dati necessari dal database:



Si aggiunge alla pagina una DB cross-tab (la pivot di FastReport), collegata alla query appena creata e configurata come in figura. E' necessario aggiungere un terzo campo a quelli relativi alla quantità che servirà solo come contenitore del valore percentuale. Nel nostro esempio è CODICE_FASE ma poteva essere scelto qualsiasi altro campo della query.

ode Data Page1

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

MasterData: MasterData1

CODICE_PARTE	QUANTITA	QUANTITA_CONSUNTIVO	CODICE_FASE
[CODICE_PARTE]	0	0	1
Grand Total	0	0	1

Cross-tab Editor

Source data

NicimQuery1

- CODICE_ORDINE
- CODICE_FASE
- CODICE_PARTE
- CODICE_VERSIONE
- CODICE_ASSIEME
- DATA_MOVIMENTO
- CODICE_COMMESSA
- CODICE_MAGAZZINO
- SEGNO

Cross-tab structure

CODICE_PARTE A-Z

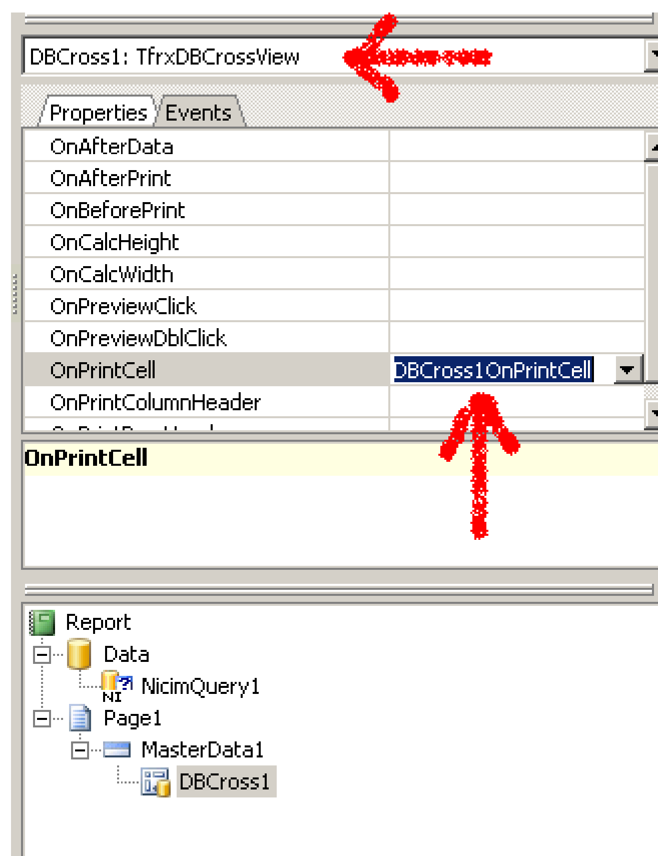
QUANTITA Sum
QUANTITA_CONSUNTIVO Sum
CODICE_FASE Count

Select style

CODICE_PARTE	QUANTITA	QUANTITA_CONSUNTI
[CODICE_PARTE]	0	
Grand Total	0	

- [x] Show title
- [x] Show corner
- [x] Column header
- [x] Row header
- [x] Column grand total
- [x] Row grand total
- [x] Auto size
- [x] Border around cells

Si seleziona nella pagina la DB cross-tab, poi sull'Object Inspector si seleziona la linguetta Events per l'oggetto cross-tab e si fa doppio click alla voce "OnPrintCell".



Nella pagina Code si aggiungono 2 variabili e si compila il metodo DBCross1OnPrintCell in questo modo:

Code | Data | Page1

Language: PascalScript

```
var
  Valore1, Valore2 : double;


procedure DBCross1OnPrintCell(Memo: TfrxMemoView; RowIndex, Column]
begin
  if CellIndex = 0 then
  begin
    if Value <> null then
      Valore1 := Value
    else
      Valore1 := 0;
  end;
  if CellIndex = 1 then
  begin
    if Value <> null then
      Valore2 := Value
    else
      Valore2 := 0;
  end;
  if CellIndex = 2 then
    Memo.Text:= FormatFloat('0.00 %', (Valore2 * 100 / Valore1));
end;

begin

end.
```

Il report è così completo. Questo è il risultato:

CODICE_PARTE			
	QUANTITA	QUANTITA_CONSUNTIVO	Percentuale
010131468	15000	6500	43,33 %
010131595	3000	0	0,00 %
010131671	4200	0	0,00 %
010131704	6000	6480	108,00 %
010131789	4000	0	0,00 %
010131932B400	800	168	21,00 %
010132033	60000	16000	26,67 %
010132189	20000	0	0,00 %
010132193	3000	0	0,00 %
010132209	22000	0	0,00 %
010132323	2200	0	0,00 %
010132329	8800	0	0,00 %
010132332	11000	0	0,00 %
010132333	20000	9500	47,50 %
0101A1789A203	2880	0	0,00 %
0101A1932A401	1680	420	25,00 %
0101A23231101	2160	0	0,00 %
0101A2361B201	20000	0	0,00 %
0101P16021201	2000	0	0,00 %
0101P16341202	600	0	0,00 %
0101P20088202	1000	0	0,00 %
Grand Total	2237432	99888	4,46 %

11.5 Come creare una pivot con percentuali calcolate sui totali

A partire da una pivot fatta così:

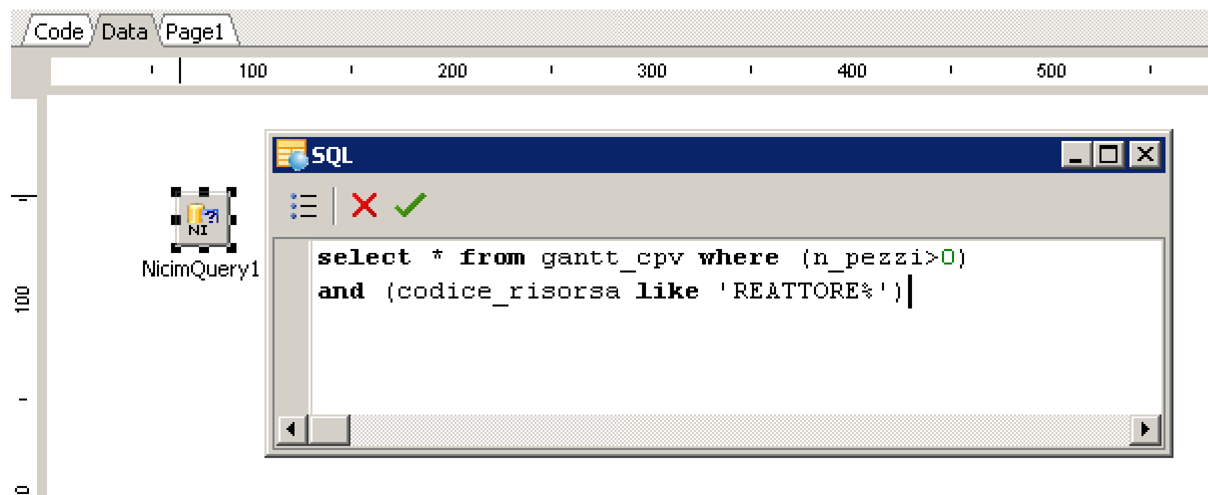
	A0	A1	A2	Totale
KEY 1	1	1	0	2
KEY 2	2	2	4	8

Vogliamo ottenere una pivot costruita così:

	A0	A1	A2	Totale
KEY 1	50%	50%	0%	2
KEY 2	25%	25%	50%	8

A partire dai dati della tabella GANTT_CPV estraiamo l'elenco dei pezzi prodotti per parte sulle risorse del gruppo REATTORI.

Aggiungiamo nella pagina Data una nuova query:



Si aggiunge alla pagina una DB cross-tab (la pivot di FastReport), collegata alla query appena creata e configurata come in figura.

MasterData: MasterData1 NicimQuery1

N_PEZZI	CODICE_RISORSA
CODICE	
[CODICE	

PageFooter:

Cross-tab Editor

Source data

NicimQuery1

- PROG
- CODICE_RISORSA
- CODICE_ORDINE
- NUMERO_FASE
- NUMERO_SOTTOFAS
- CODICE_FASE
- CODICE_PARTE
- CODICE_VERSIONE
- CODICE_ASSIEME

Cross-tab structure

CODICE_RISORSA A-Z

CODICE_PARTE A-Z

N_PEZZI Sum

Select style

N_PEZZI	CODICE_RISORSA	
CODICE_PARTE	[CODICE_RISORSA]	**Grand Total**
[CODICE_PARTE]	0	0

- [x] Show title
- [x] Show corner
- [x] Column header
- [x] Row header
- [x] Column grand total
- [ ] Row grand total

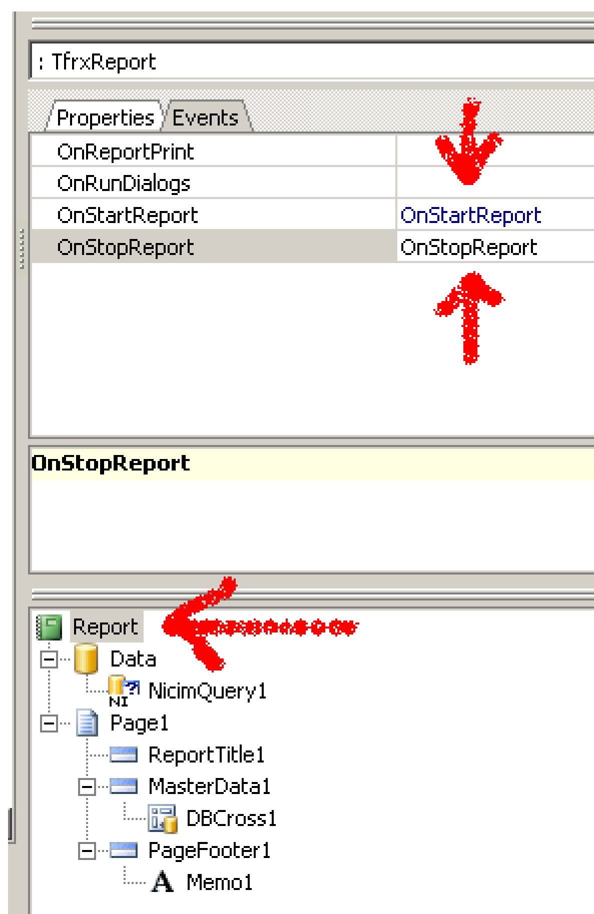
- [x] Auto size
- [x] Border around cells
- [ ] Print down then across
- [x] Reprint headers on new page
- [ ] Side-by-side cells
- [ ] Join equal cells

Nelle opzioni del report (Report->Options) si checca la proprietà "Double pass".

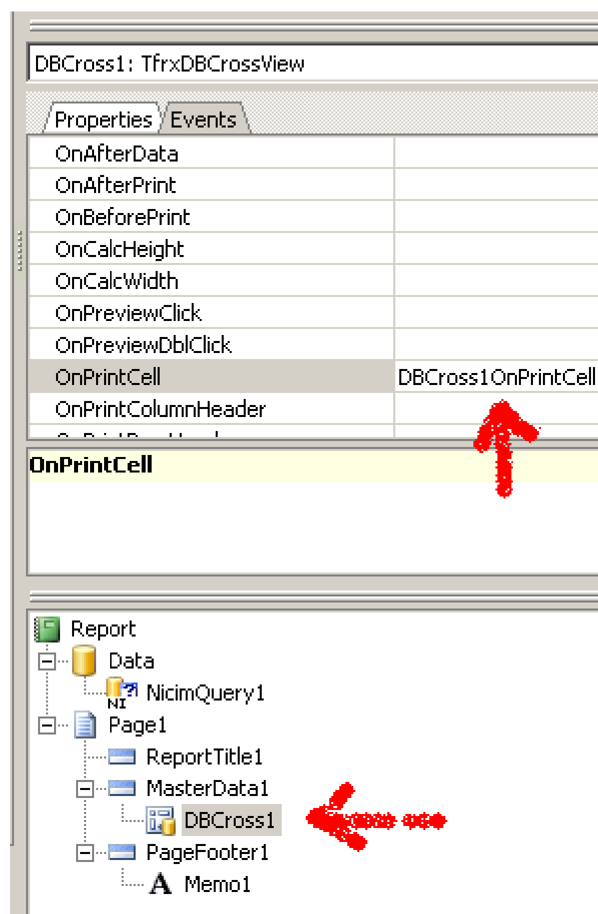
Si passa alla pagina "Code" del report.
Qui va aggiunto in alto il codice:

```
var
  List : TStringList;
```

Nella pagina del report, attraverso l'Object Inspector, si seleziona l'oggetto "Report" e nella linguetta Events si fa un doppio click sull'evento OnStartReport e sull'evento OnStopReport:



Sempre attraveso l'Object Inspector, si seleziona l'oggetto DBCross e si fa doppio click sul suo evento OnPrintCell:



Nella pagina "Code" si riempiono gli eventi appena creati con questo codice:

Code Data Page1

Language: PascalScript

```
var
  List : TStringList;



procedure OnStartReport(Sender: TfrxComponent);
begin
  List := TStringList.Create;
end;

procedure OnStopReport(Sender: TfrxComponent);
begin
  List.Free;
end;

procedure DBCross1OnPrintCell(Memo: TfrxMemoView; RowInde
begin
  if (not Engine.FinalPass) and
  (DBCross1.IsGrandTotalColumn(ColumnIndex)) then
     begin
       if Value <> null then
         List.Add(FloatToStr(Value))
       else
         List.Add('0');
     end;
  if (Engine.FinalPass) and
   (not DBCross1.IsGrandTotalColumn(ColumnIndex)) and
   (Value<>null) then
     Memo.Text:=FormatFloat('0.00 %',
       Value/StrToFloat(List.Strings[RowIndex])*100);
end;

begin

end.
```

Questo è il risultato:

N_PEZZI	CODICE_RISORSA					
CODICE_PARTE	REATTORE-1	REATTORE-2	REATTORE-4	REATTORE-5	REATTORECHEL-01	**Grand Total**
010130829	40,00 %			60,00 %		55000
010130902			100,00 %			9400
010130907		100,00 %				77000
010130968				100,00 %		12501
010130971	66,67 %		33,33 %			39000
010130973			50,00 %	50,00 %		22000
010131266			100,00 %			6000
010131411			100,00 %			12000
010131468	100,00 %					8500
010131595	100,00 %					3000
010131704					100,00 %	16,6499999614
010132189		100,00 %				20000
010132193				100,00 %		3000
010132209	100,00 %					22000

11.6 Come creare una pivot con header colorati secondo una condizione

In questo esempio vogliamo modificare una pivot in modo da evidenziare alcune celle tra quelle delle instestazioni delle righe in base ad una condizione sul loro valore.

In particolare, partendo da questa pivot:

1 Close

TEMPO_LAV_SIMULATO		DATA_ORA_INIZIO_SIMULATA2		
CODICE_RISORSA	NUMERO_FASE	19/03/2009 17.29.27	19/03/2009 17.32.27	19/03/2009 18.08.10
CDL10	10	4,8		
	Total	4,8	0	0
CONF-1KG	5	5,33333333333332		
	10			
	Total	5,33333333333332	0	0
CONF-GRANULARI	10	70,3313333263002		
	10,1			
	10,11			
	10,2			
	Total	70,3313333263002	0	0
CONF-MANUALE	10	15,1	12,64	
	20			
	Total	15,1	12,64	0
CONF-ROVEMA	10	26,666668		
	10,1			
	10,2			
	20			
	Total	26,666668	0	0
DECANTATORE-2	41	206,454545454		
	Total	206,454545454	0	0

vogliamo ottenere una pivot in cui le intestazioni del campo NUMERO_FASE vengano dipinte in rosso quando il NUMERO_FASE è maggiore di 10:

1 Cancel

TEMPO_LAV_SIMULATO		DATA_ORA_INIZIO_SIMULATA2		
CODICE_RISORSA	NUMERO_FASE	19/03/2009 17.29.27	19/03/2009 17.32.27	19/03/2009 18.08.10
CDL10	10	4,8		
	Total	4,8	0	0
CONF-1KG	5	5,33333333333332		
	10			
	Total	5,33333333333332	0	0
CONF-GRANULARI	10	70,3313333263002		
	10,1			
	10,11			
	10,2			
	Total	70,3313333263002	0	0
CONF-MANUALE	10	15,1	12,64	
	20			
	Total	15,1	12,64	0
CONF-ROVEMA	10	26,666668		
	10,1			
	10,2			
	20			
	Total	26,666668	0	0
DECANTATORE-2	41	206,454545454		
	Total	206,454545454	0	0
DECANTATORE-5	41	0,05		
	Total	0,05	0	0
	10	36,42857142777		

La funzione di Highlight disponibile in FR non funziona sulla celle delle intestazioni di righe o colonne di una pivot, ma solo sulle celle che contengono i dati aggregati (nell'esempio la colonna TEMPO_LAV_SIMULATO).
Bisogna quindi seguire un'altra strada.

Dopo aver selezionato la DBCrossView nell'editor di FastReport, si controlla in quale posizione si trova la colonna NUMERO_FASE nella pivot.
Nel nostro esempio la colonna NUMERO_FASE è la seconda colonna delle instestazioni delle righe (la prima è CODICE_RISORSA), quindi ha indice 1 (l'indice parte da zero).
Mantendo sempre selezionata la crossview, si seleziona la linguetta Events e si fa doppio click a fianco dell'evento OnPrintRowHeader:

All'interno dell'editor del codice, si riempie la funzione appena creata con queste istruzioni:

Code | Data | Page1

Language: PascalScript

```
procedure DBCross1OnPrintRowHeader(Memo: TfrxMemoView; HeaderIndexe
var
  Temp : string;
begin
  Temp := VarToStr(HeaderValues[1]);
  if ValidFloat(Temp) then
  begin
    if StrToFloat(Temp) > 10 then
      Memo.Color := clRed;
  end;
end;

begin

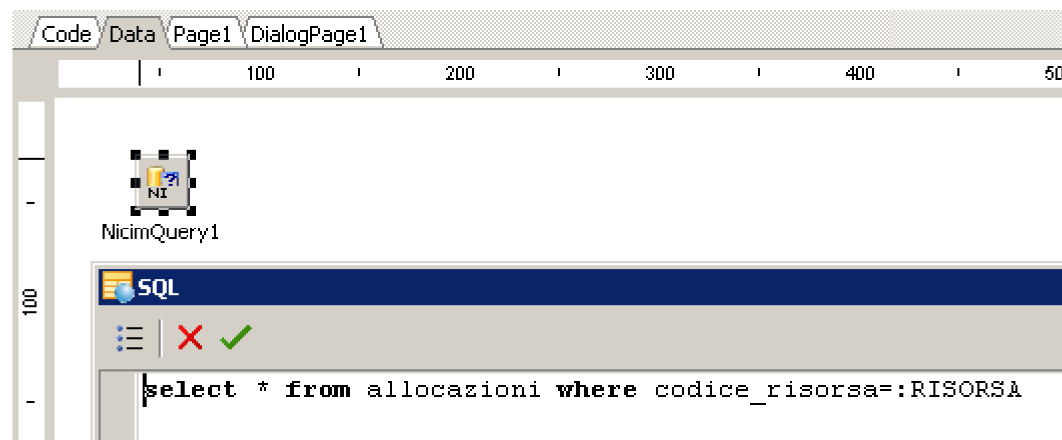
end.
```

HeaderValues[1] restituisce il valore della celle di testata della riga alla posizione 1, abbiamo visto sopra che l'indice 1 è quello della colonna NUMERO_FASE. Se avessimo avuto bisogno di far riferimento alla colonna CODICE_RISORSA, allora avremmo dovuto usare HeaderValues[0].

11.7 Come creare un report con una dialog

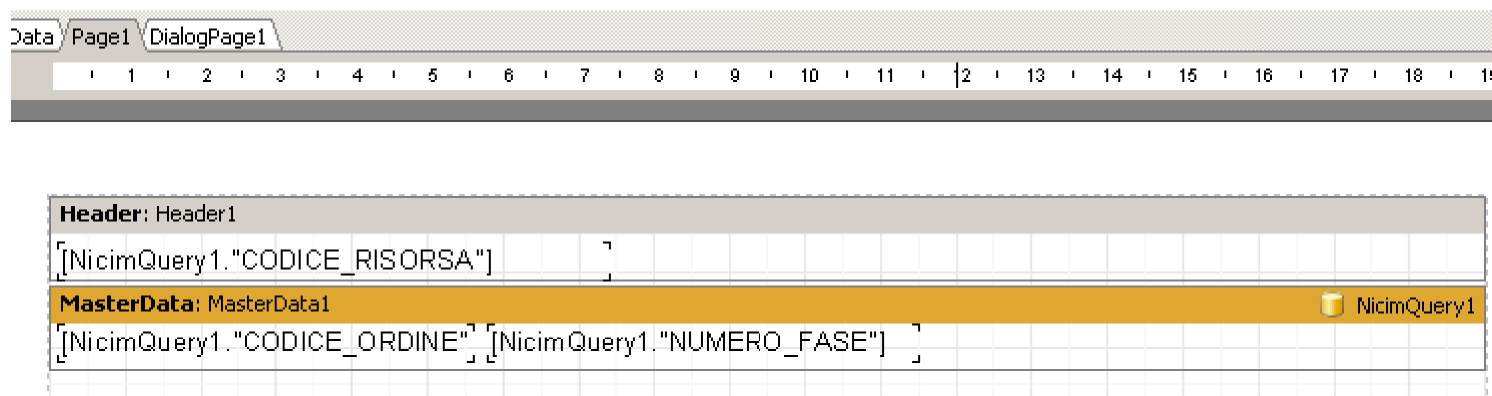
Vediamo come creare un report con una dialog che alla partenza permetta all'utente di valorizzare un parametro in una query e quindi di pilotare i dati visualizzati nel report di conseguenza.

Creiamo un nuovo report e, nella pagina Data, aggiungiamo una query chiamata NicimQuery1 a cui aggiungiamo l'SQL visibile in figura:



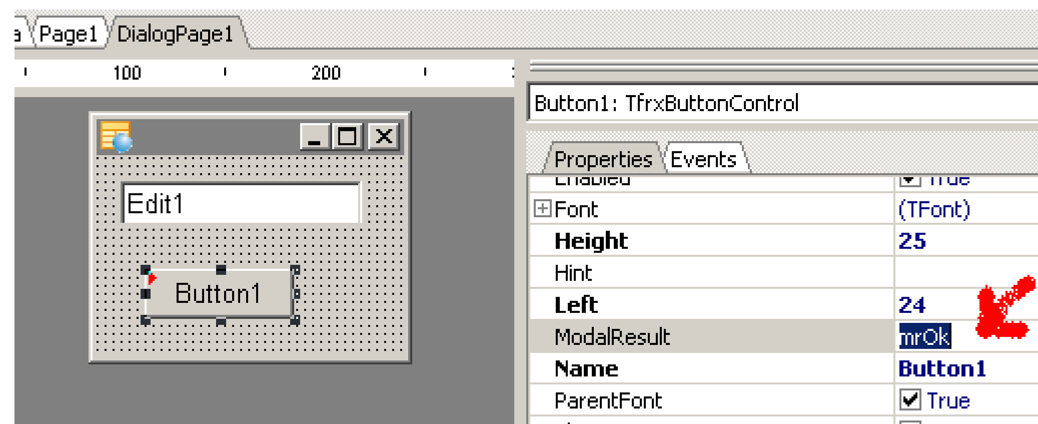
Aggiungiamo al report un Header ed una sezione MasterData collegata alla NicimQuery1.

Vogliamo che nell'header della pagina compaia il campo CODICE_RISORSA della query e che il resto del report contenga l'elenco delle fasi allocate su quella risorsa (campi CODICE_ORDINE e NUMERO_FASE estratti dalla query):

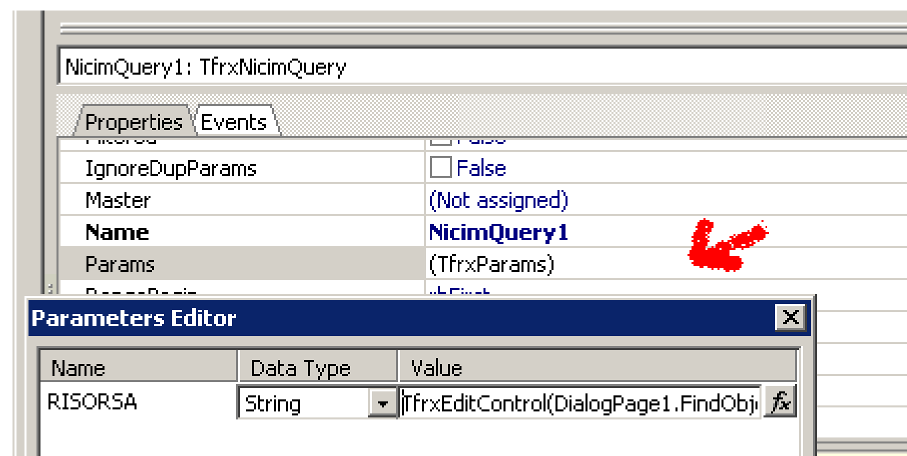


Aggiungiamo al report una nuova dialog (menu File -> New Dialog) su cui trasciniamo un Edit control e un Button control.

Impostiamo la proprietà ModalResult del pulsante a mrOk.



Torniamo nella pagina Data, selezioniamo la query e nelle proprietà Params impostiamo il DataType del parametro ed il suo valore come in figura:



La formula utilizzata per il valore è questa:

TfrxEditControl(DialogPage1.FindObject('Edit1')).Text

Se ora eseguiamo il report il risultato è il seguente:

- viene visualizzata la finestra di dialogo in cui l'utente puo' inserire un CODICE_RISORSA



- al click del pulsante, il report visualizzerà le fasi allocate sulla risorsa prescelta

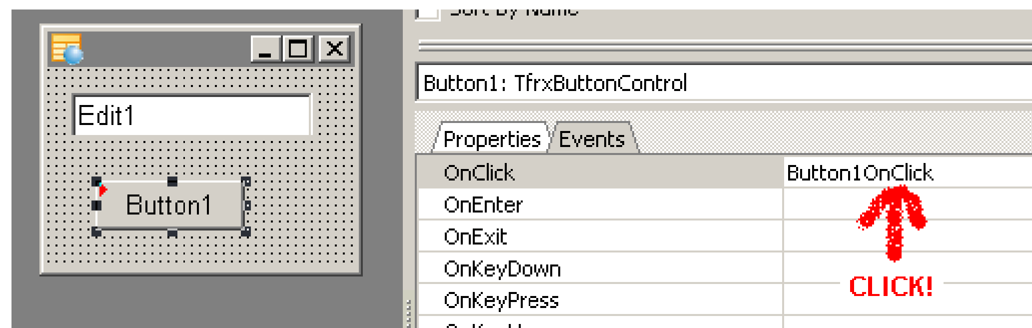
ESSICCATORE

200600045385	10
200600047041	10
200600047047	10
200600027739	10
200600028772	30
200600030135	10
200600030140	10
200600030164	10
200600030165	10
200600030168	10
200600030171	30
200600030463	10
200600039203	30
200600039205	30

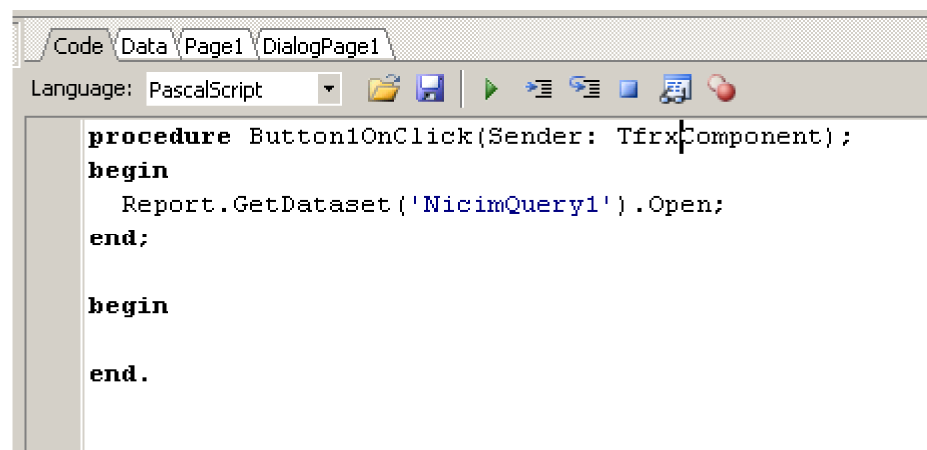
Attenzione: potrebbe accadere che la colonna della query visualizzata nell'header della pagina (CODICE_RISORSA nell'esempio) non venga visualizzata correttamente.

Se accadesse, possiamo risolvere la situazione in questo modo:

- nella pagina di configurazione della Dialog selezioniamo il pulsante e facciamo doppio click sull'evento OnClick nella linguetta Events del pannello delle proprietà dell'oggetto:



- aggiungiamo il codice qui illustrato nel corpo dell'evento appena creato:




```
procedure Button1OnClick(Sender: TfrxComponent);
begin
  Report.GetDataset('NicimQuery1').Open;
end;

begin

end.
```

Questo dovrebbe bastare a far funzionare anche l'header del nostro report.

Parte XII

12 Varie

12.1 Parametri da command line

AtomosVision supporta diversi parametri da command line per abilitare diverse opzioni automaticamente.

L'elenco aggiornato dei parametri e la loro descrizione è ottenibile lanciando l'applicativo da command line con il parametro HELP:

```
AtomosVision.exe HELP
```

12.2 Si connette con BDE o altro?

Per verificare quale tecnologia è utilizzata da Vision per connettersi al database principale si può seguire la seguente procedura:

- lanciare Vision da command line con il parametro DEBUGLOG
- alla partenza di Vision comparira' una finestra di debug che scivolera' in secondo piano sotto la finestra principale del programma
- connettersi al db come al solito o aspettare che lo faccia da solo Vision usando il .CON di default
- far uscire Vision dallo stato di finestra massimizzata e ridimensionarlo quel tanto che basta per acciuffare la finestra di debug che sta sotto
- nella finestra di debug bisogna scorrere un po' le righe partendo dall'alto. Ce ne sono un certo numero che iniziano con la stringa "AtomosDbControllers [INFO]" seguite da una data/ora e dalla stringa "##### db alias report start ####"
- cercare il blocco che fa riferimento al db principale (probabilmente e' il secondo). Li' si troverà una riga contenente la stringa "Used db connection type id:" che dice quale tecnologia è utilizzata per la connessione, il valore sarà BDE (Borland Database Engine) oppure ODAC (nativo ORACLE) o SDAC (nativo MS SQL Server) o altro.

Index

- A -

- B -

- C -

- D -

- E -

- F -

- G -

- H -

- I -

- J -

- M -

- P -

- Q -

- S -

- T -

http://www.atomos.it